

User Manual

WinCE manual for OEM



INFORMATION, it helps users with more details about the topic and failure to follow may lead to unpredictable results.



WARNING, Failure to follow may lead to minor injury or damage / malfunctioning of equipment



DANGER, Failure to follow may lead to injury or fatal accident to operating personal or damage/malfunctioning of equipment



CAUTION, Failure to follow may lead to malfunctioning of equipment, damage or repair



Protective Earth



DC Supply

PREFACE

Original equipment manufacturer reserves the right to change information available in this document without notice. Original Equipment manufacturer is not liable for any damages incurred to equipment/personal during installation or use of equipment as explained in this document. User must acquire sufficient knowledge & skills prior to use the equipment in the application and follow all the local standards & regulations to meet safety requirements

TRADE MARK ACKNOWLEDGEMENTS

WinCE[®] is registered trade mark of Microsoft Corporation

Wonderware[®] is registered trade mark of Invensys

Indusoft[®] is registered trade mark of Indusoft corporation

Movicon[®] is registered trade mark of Progea

TABLE OF CONTENTS

1.	INTRODUCTION	4
1.1	WINCE IMAGE VARIANTS.....	4
1.2	WINCE CORE	4
1.3	WINCE PROFESSIONAL	5
1.4	SYSTEM MEMORY	6
1.5	OEM OPTIONS	8
2.	OEM TOOLS	9
2.1	OEM TOOLS	9
2.1.1	AUTO RUN MAKER	9
2.1.1.1	How to start application at HMI Power ON	10
2.1.1.2	How to select a boot logo and desktop logo	15
2.1.1.3	How to use Registry editor	21
2.1.1.4	How to create shortcut at Desktop & Program.....	31
2.1.1.5	How to run <i>Indusoft</i> application in HMI	36
2.1.1.6	How to run <i>Wonderware</i> application in HMI.....	43
2.1.1.7	How to run <i>Movicon</i> application in HMI	51
2.1.1.8	Reserved	60
2.1.1.9	Reserved	60
2.1.1.10	How to view HMI screen in PC via VNC	61
2.1.2	BACKLIGHT ADJUSTMENT	65
2.1.3	TOUCH PANEL CALIBRATION	67
2.1.4	UPGRADE WINDOWS CE IMAGE	71
2.2	WINCE CORE & PROFESSIONAL COMPARISION	74
2.3	APPENDIX	81
2.4	WINCE IMAGE CONVERSION.....	82
2.4.1	Control Center V1.1x to WinCE Professional	82
3.	BOARD SUPPORT PACKAGE (BSP)	88
3.1	SOFTWARE REQUIREMENTS	88
3.2	INSTALL BSP SMDK6410	88
3.3	INSTALL BSP SMDKC100	96
3.4	ADD "USER ACCOUNT" INTO PROJECT	104
3.5	USE "IMGPAK.EXE" TO BUILD OEM_6410.BIN (OEM_PC100.BIN)	113
3.6	HOW TO CHANGE THE BOOT LOGO.....	116
3.7	BUZZER CONTROL.....	123
3.8	TOUCH CALIBRATION	126
3.9	RETRIEVE MAC ADDRESS.....	127
3.10	CALL BACKLIGHT CONTROL DIALOG.....	128
4.	VISUAL STUDIO SAMPLE	129
5.	FAQ	140

1. INTRODUCTION

All our HMI are available with WinCE 6.0 operating system. WinCE is compact operation system from Microsoft. We supply open WinCE 6.0 platform and it provides flexibility for the OEM's to run WinCE based SCADA or customized software in HMI.

1.1 WINCE IMAGE VARIANTS

Following WinCE images are available

HMI model	WinCE version	File name
HMI 450,730	WinCE Core	OEM_6410
HMI 450,730	WinCE Professional	OEM_6410
HMI 750/1050/1550	WinCE Core	OEM_PC100
HMI 750/1050/1550	WinCE Professional	OEM_PC100

1.2 WinCE Core

By default, WinCE Core operating system preinstalled in HMI.

If you select ordering code 1 or 2, then, WinCE Core operating system + Control Center firmware is preinstalled. In this case, it is possible to use our HMI editing software for application development

In case if you wish to run your own applications on WinCE Core, you need to replace the Windows CE image such that at Power ON, it directly goes to Windows CE desktop. Contact the factory for the WinCE images

For running third party applications, we suggest to please first check the difference between WinCE Core and WinCE Professional and then decide about version required

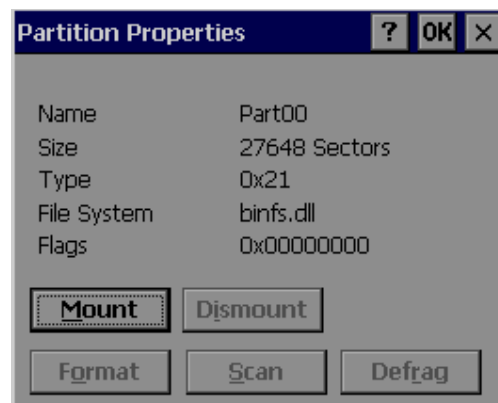
Approx. image size for WinCE Core: 40 MB

1.3 WinCE Professional

In addition to WinCE Core, some additional features are available in WinCE Pro. Please refer documentation from Microsoft for exact differences between WinCE Core and WinCE Professional Versions. However, to limit the WinCE image size, not all features are enabled. Some major features in Windows CE Professional version includes

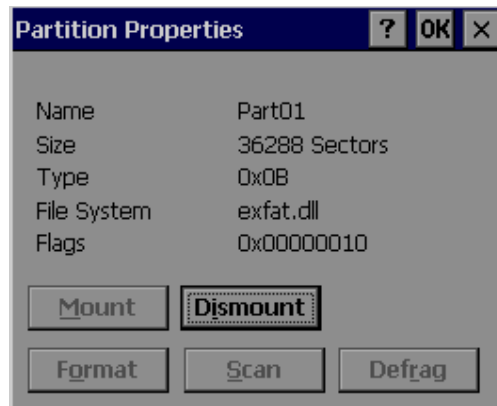
1. Web browser (Internet explorer)
2. Word pad text editor
3. Java and Visual basic scripts etc.
4. VNC Server
5. VNC Client
6. Remote desktop connection etc.

Approx. image size for WinCE Professional: 55 MB



Each Sector is 2 bytes

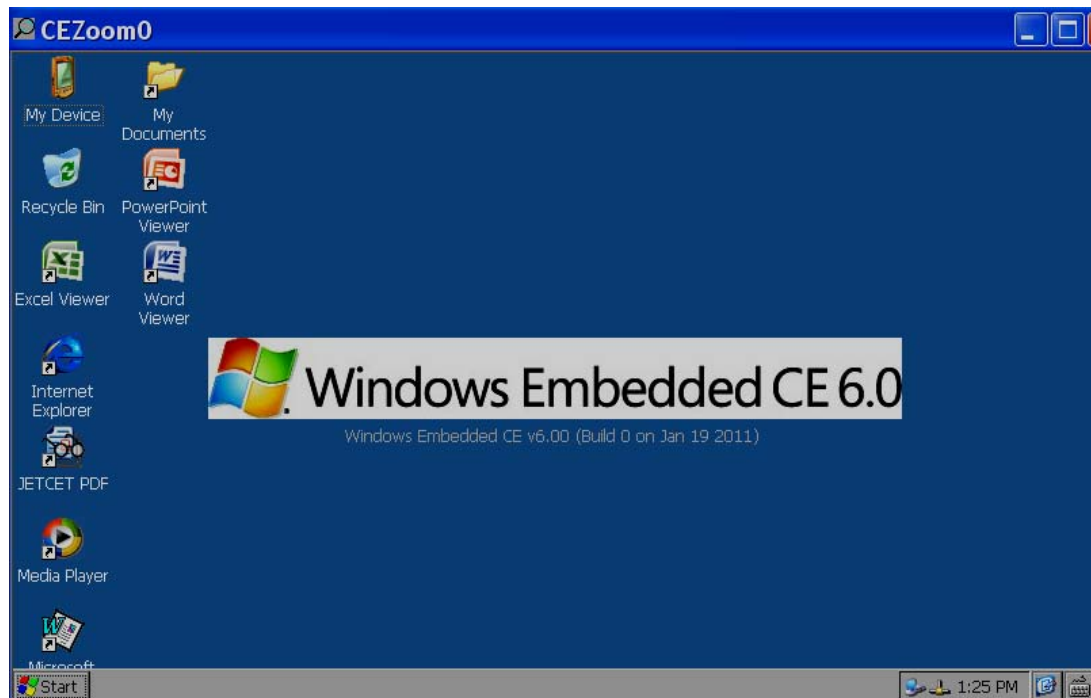
So, $27648 \times 2 = 55296$ bytes, approx: 55 MB



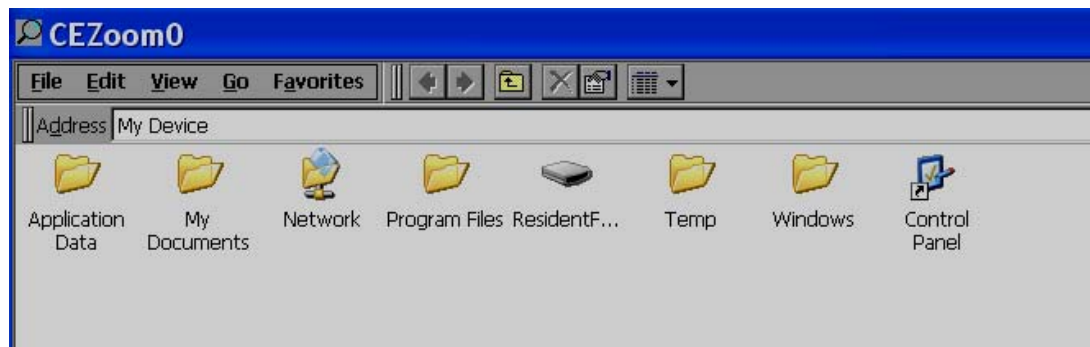
1.4 SYSTEM MEMORY

The system memory (RAM) divided into two parts

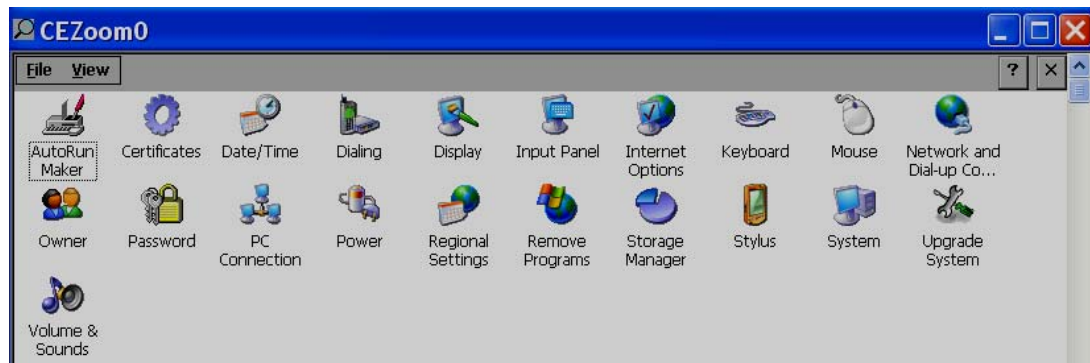
Storage Memory & Program memory



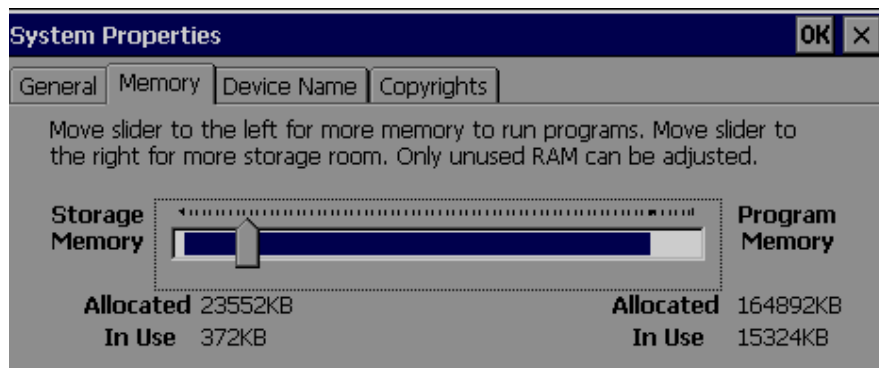
Open "My device" (Press two times quickly on top of "My Device" icon)



Open "Control Panel"



Open "System" and then open "Memory" tab



Free Prog. Memory = Allocated Prog. Memory – Prog. Memory in use

In above example, 164892-15324=149568 KB

Approx. 146 MB free

1.5 OEM OPTIONS

OEM Core/OEM Pro image

If ordering code is 0, then, we will load standard WinCE Core or WinCE Professional image as per your requirement. It is possible to run third party applications directly using this image. You can use readily available tools for starting application at HMI Power ON, boot Logo etc.

Board support package (BSP):

It is required for companies who are having professional engineers with skills at WinCE. In this case, they can generate their own WinCE images. Please contact the factory/supplier for BSP files

Software development kit (SDK)

In case if you would like to run Visual Studio applications in HMI, then, we will supply SDK. It should be used with project and build the application, then download to HMI. Please contact the factory/supplier for SDK file

2. OEM TOOLS

2.1 OEM TOOLS

These are already available in WinCE image and can be accessed from “Control Panel” in Windows

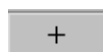
2.1.1 AUTO RUN MAKER



Source Folder: It is to select the location for the source folders/files to be copied from USB disk to Resident flash

Create directory and Copy folder: Select this option if you wish to create a default directory at Resident flash by name “AutoRunMaker” and then copy all the contents to this folder

Copy folders directly to Resident Flash: Select this option if you wish to copy the folders/files directly to Resident flash



Press this to add the folder



Press this to remove the folder. If there are some folders created earlier, it will show the path, first select required path to be removed and then press this button to remove the content from Resident flash

Startup: It is to select the path for the exe file at HMI Power ON

Logo: It is to select the boot logo. This logo will appear after HMI Power ON

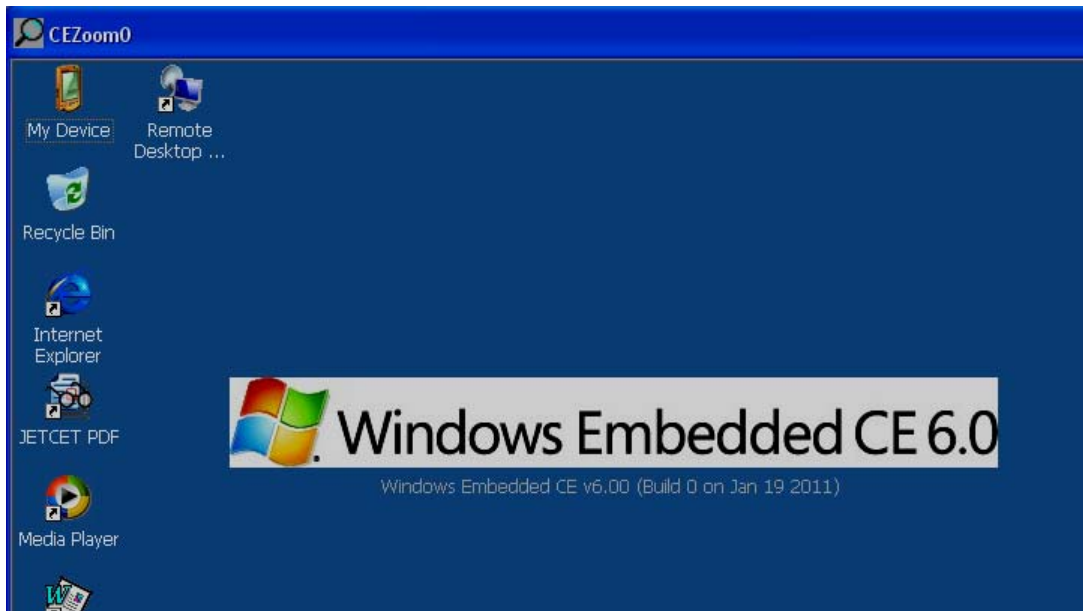
Register: It is to add/modify contents to Windows Registry

Shortcut: It is to create desktop shortcut for the required applications

2.1.1.1 How to start application at HMI Power ON

Step-1: Prepare a folder by TEST at USB disk and copy all the required HMI/SCADA run time files including xxx.exe into above folder. The contents will be shown as below

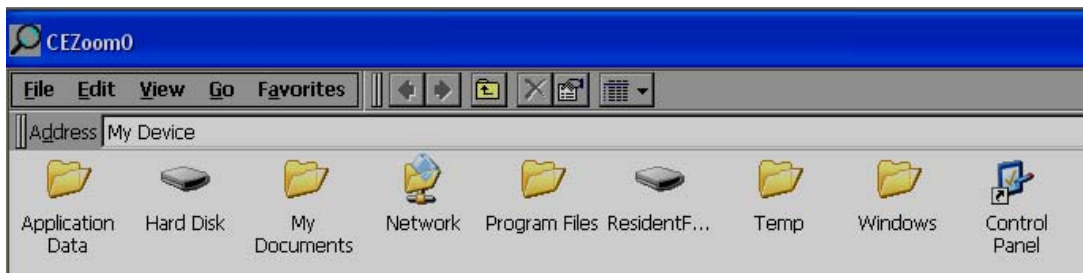
Step-2: Insert USB disk in HMI and Power ON HMI



Step-3: Press “My Device”, first, it will be selected and it shows with blue color mark as shown below



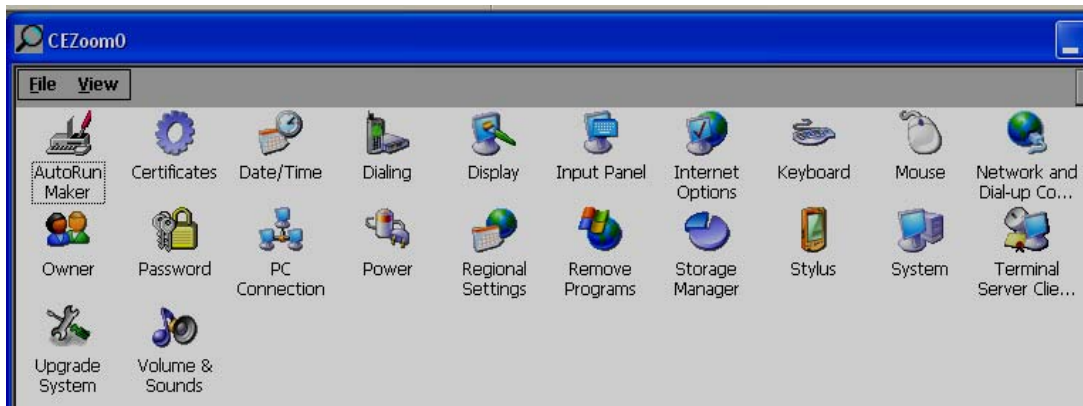
Now, press “My Device” quickly two times to open this folder



Note:

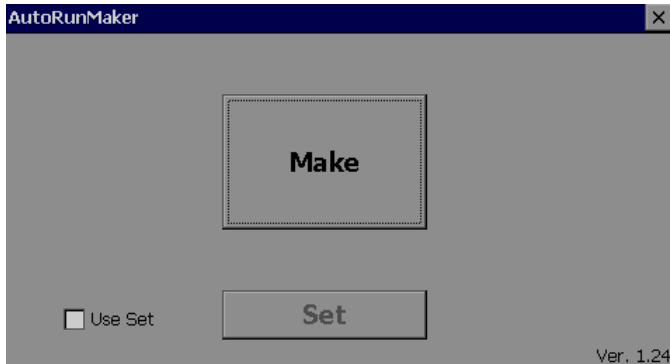
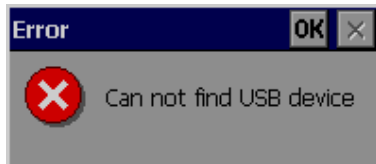
USB stick will be shown at “Hard Disk” at “My device”
Onboard flash memory will be shown as “Resident Flash” at “My device”

Select "Control Panel" and open "Control Panel"



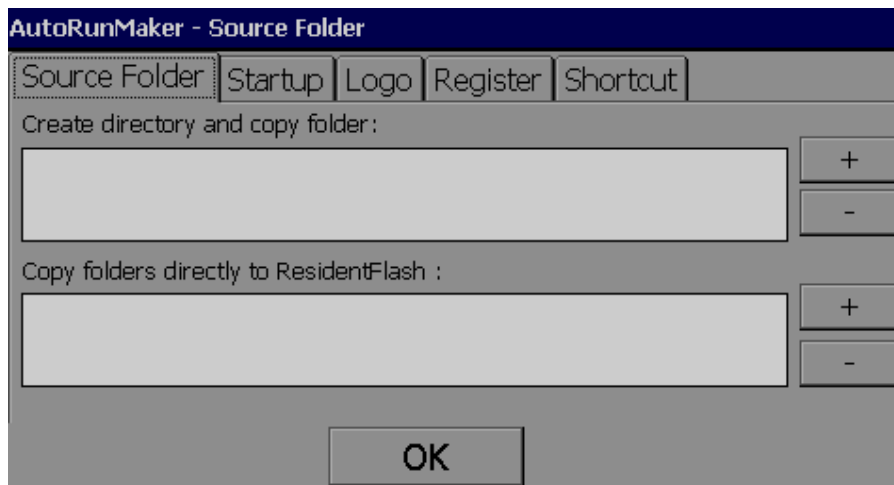
Select "AutoRun Maker" and open "AutoRun Maker"

If there is no USB device, it shows following message




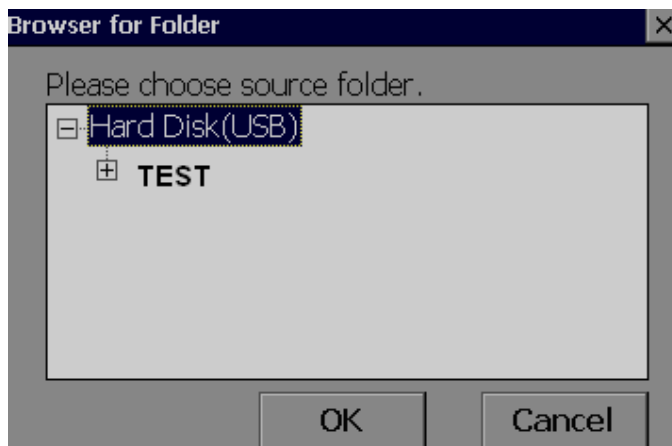
Note: Bottom Right corner shows current version of WinCE image

Select "Use Set" check box and press "Set"



Now, it is possible to copy TEST folder from USB to Resident flash

Press  button at "Copy folders directly to Resident flash"

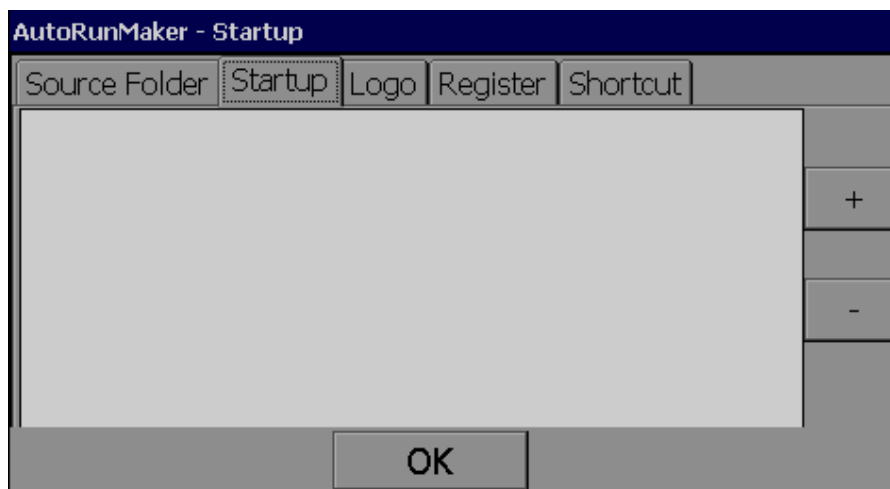


Select "TEST" and press "OK"



All the content from TEST folder at USB disk is now copied to Resident flash and now, it appears the source path as shown above.

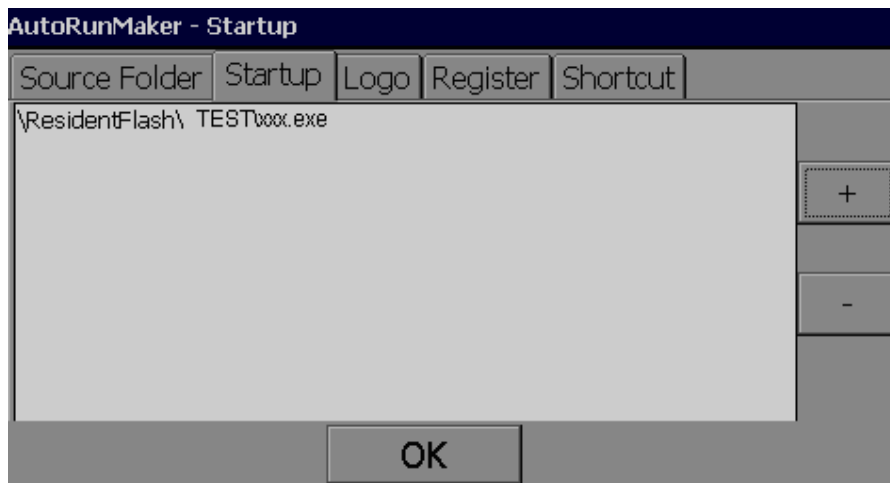
Now, press "StartUp" Tab



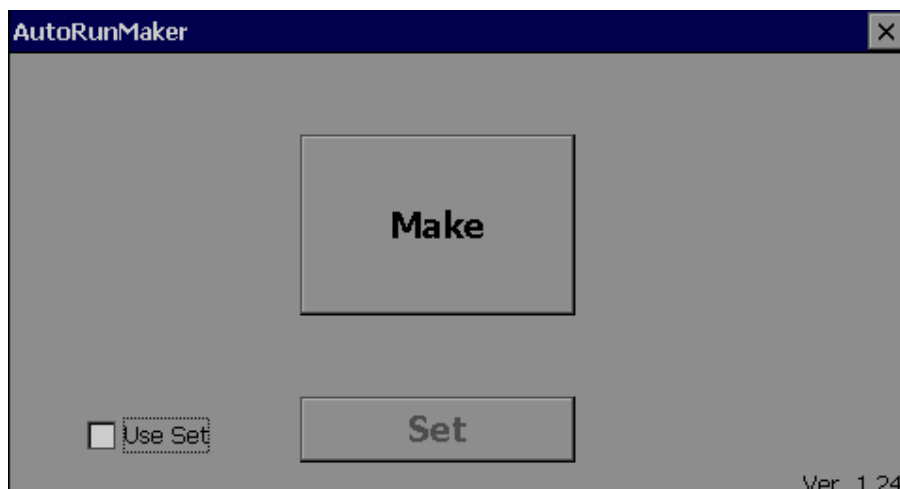
Press  button to select the path for the exe file

Select ResidentFlash\Test and open it

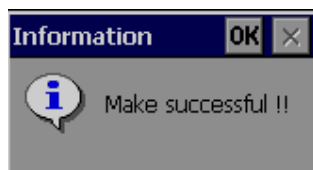
Select "xxx.exe" and then press "OK"



Press "OK"



Press "Make" button



Open "MyDevice\ResidentFlash\TEST" and double check the contents

Now, remove the USB disk

Switch OFF Power supply to HMI

Power ON HMI again

If xxx.exe file is selected to start up, then, it will be executed automatically at HMI Power

2.1.1.2 How to select a boot logo and desktop logo

Step-1: Prepare your jpg files

Image resolution

HMI size	Resolution
HMI 4.3"	480x272
HMI 7"	800x480(7")
HMI 10"/15"	1024x768

Please use the windows tool "Paint.exe" to save as a standard jpg format file.

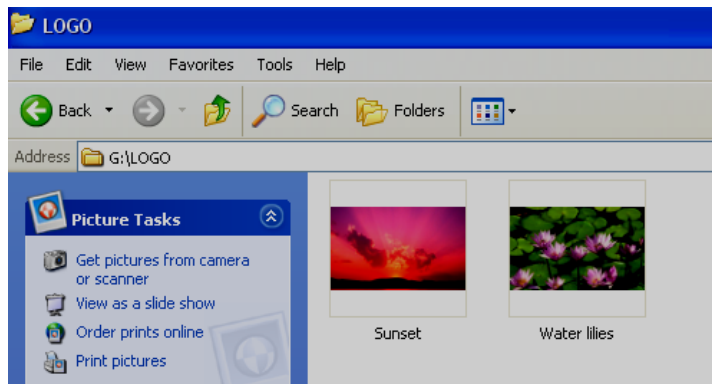
For this manual, two files are used

Sunset: for BootLogo
Water lilies for Desktop Logo

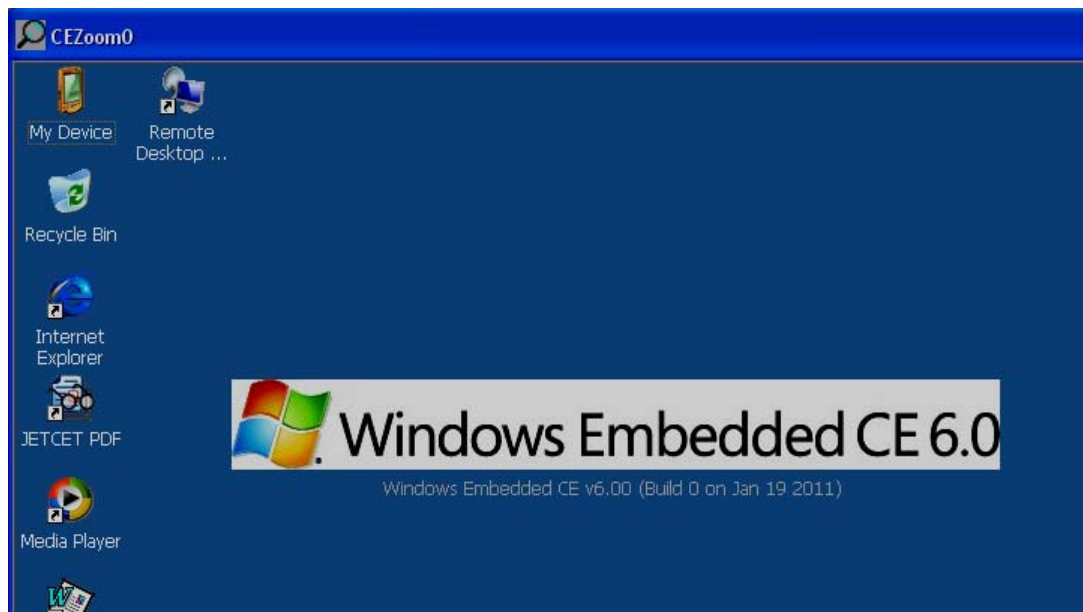
The Boot Loader can not support all JPG formats, it is recommended to use MS Paint Save as a new file in JPG format.

If the OEM Loader JPG File cause any problems and can not boot, please short COM1 pin2 and pin3 and reboot.

Copy Sunset.JPG and Water lilies.JPG to "LOG" folder USB disk



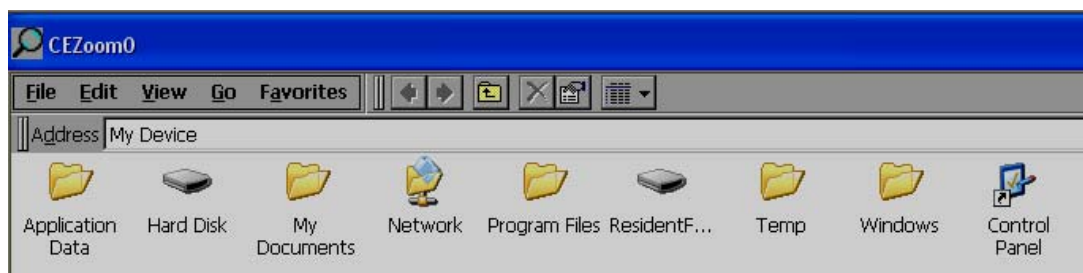
Step-2: Insert USB disk at HMI and then Power ON HMI



Press “My Device”, first, it will be selected and it shows with blue color mark as shown below



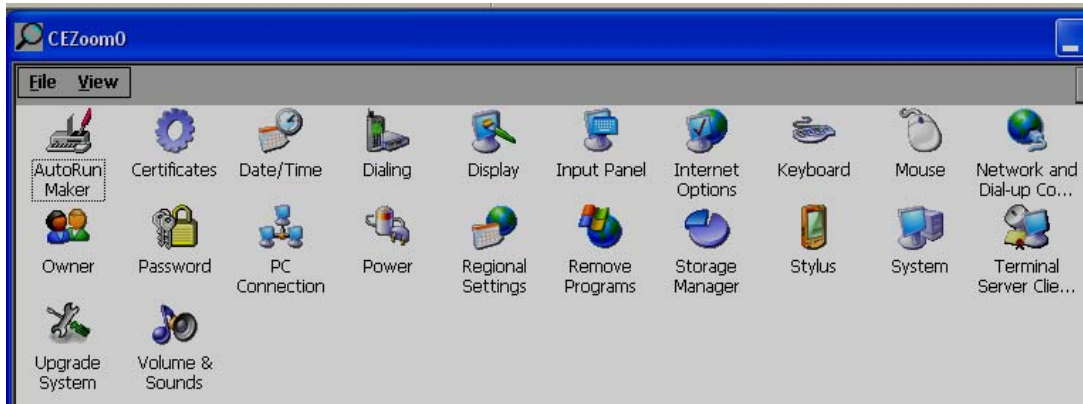
Now, press “My Device” quickly two times to open this folder



Note:

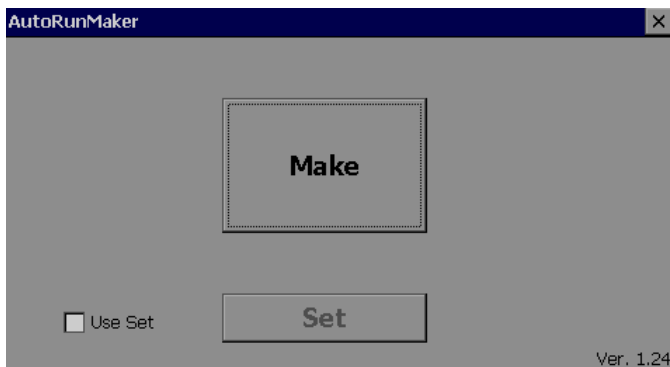
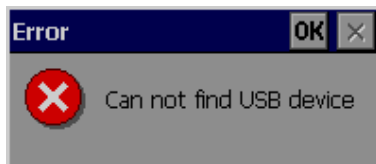
USB stick will be shown at “Hard Disk” at “My device”
Onboard flash memory will be shown as “Resident Flash” at “My device”

Select “Control Panel” and open “Control Panel”



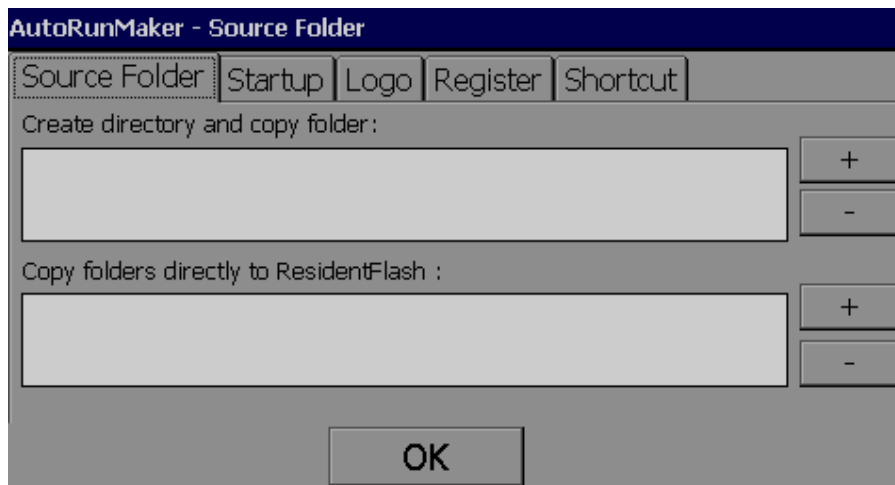
Select “AutoRun Maker” and open “AutoRun Maker”


If there is no USB device, it shows following message

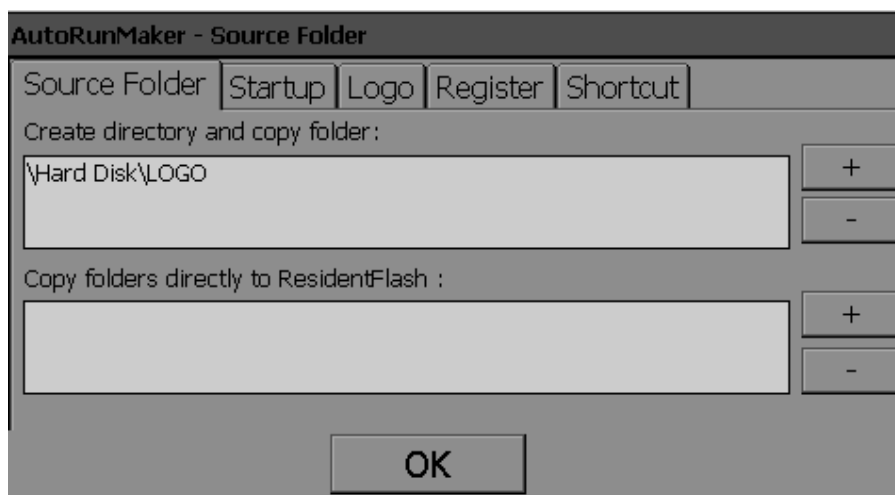
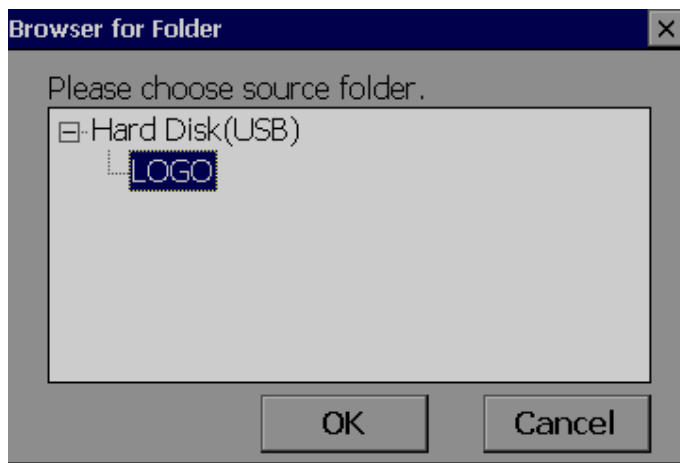


Note: Bottom Right corner shows current version of WinCE image

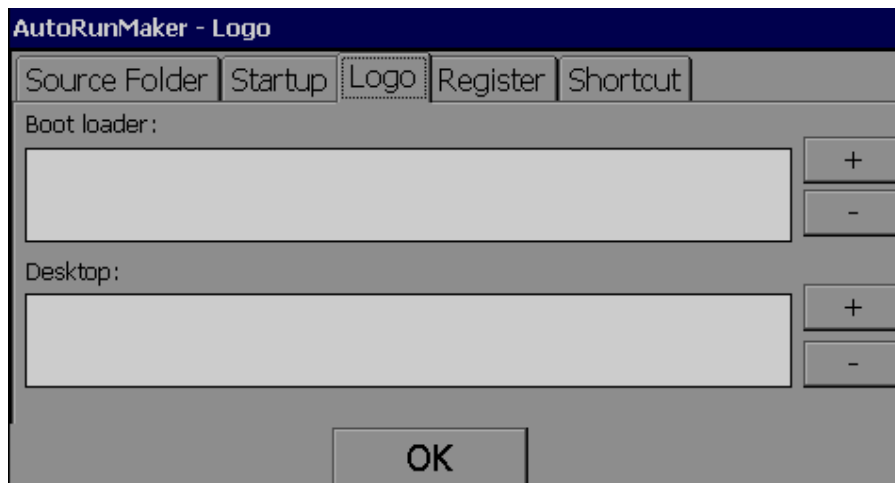
Select “Use Set” check box and press “Set”



At “Source Folder” Tab, press  and then select “LOGO” folder, then press “OK”





Press “Logo” Tab



Boot Loader: It is to select the logo to appear at HMI boot up time

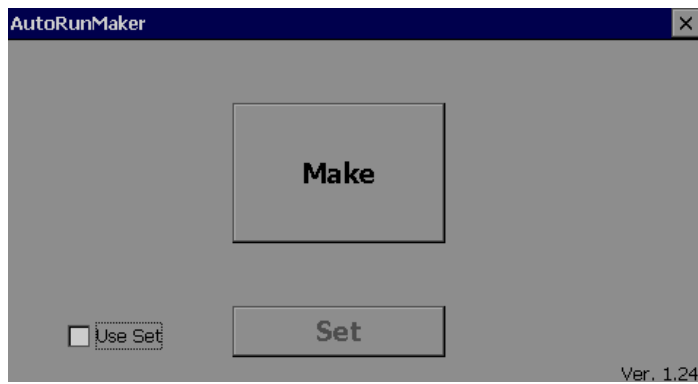
Desktop: It is to select the logo for the desktop

Press  at Boot Loader and select "Sunset" JPG file from "LOGO" folder from Resident Flash

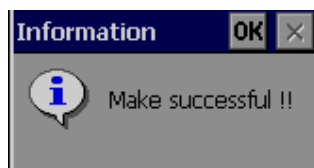
Press  at Desktop and select "Water lilies" JPG file from "LOGO" folder from Resident Flash



Press "Ok"




Press “Make”

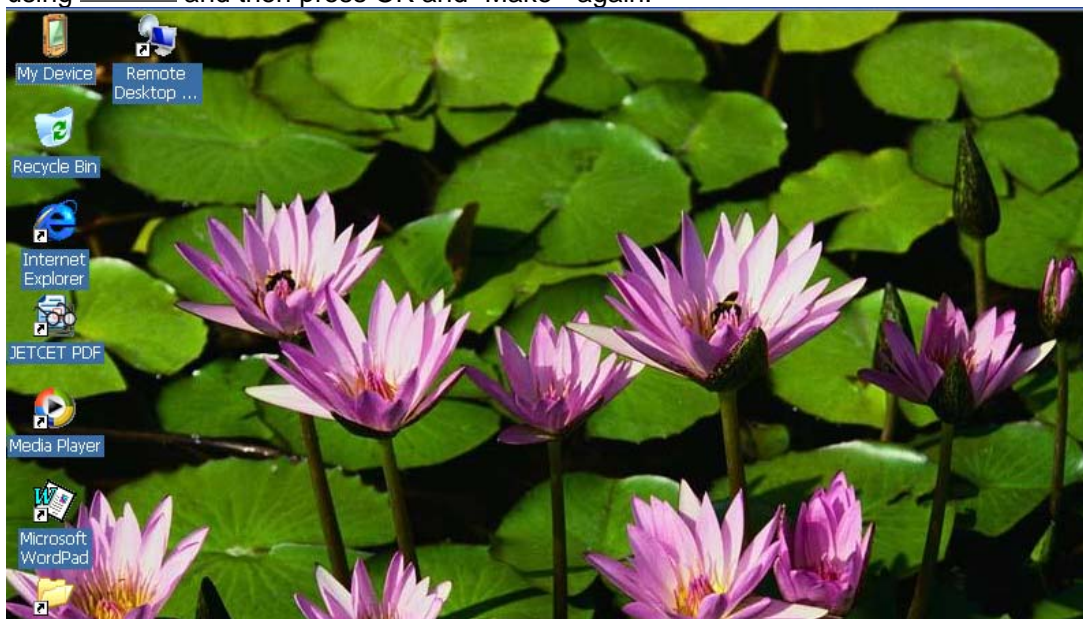


Switch Off power supply to HMI and Power ON again

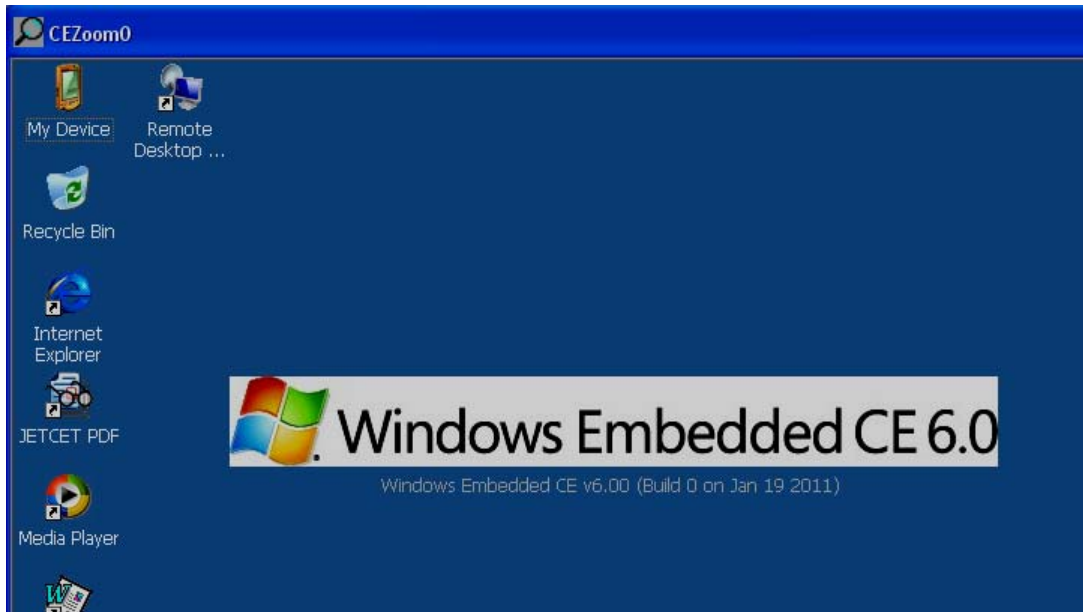
At boot loading, it shows SunSet.JPG as background

Then, it shows Water lilies.JPG in HMI desktop as shown below

Note: In case if you wish to remove the above logo and go back to default status, just remove them from the Boot Loader and Desktop path at “Logo” Tab at Auto run maker using  and then press OK and “Make “ again.



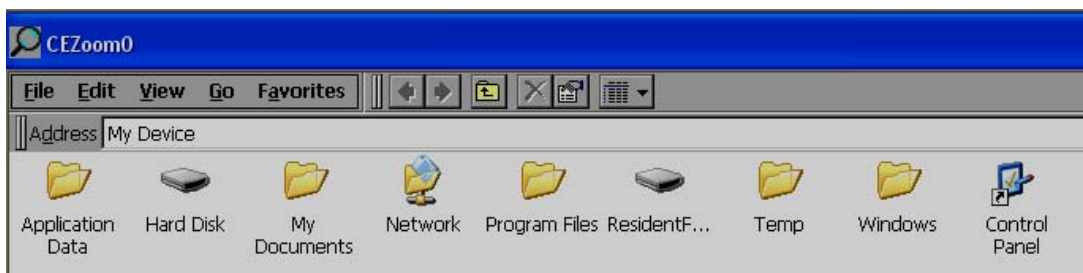
2.1.1.3 How to use Registry editor



Press “My Device”, first, it will be selected and it shows with blue color mark as shown below



Now, press “My Device” quickly two times to open this folder

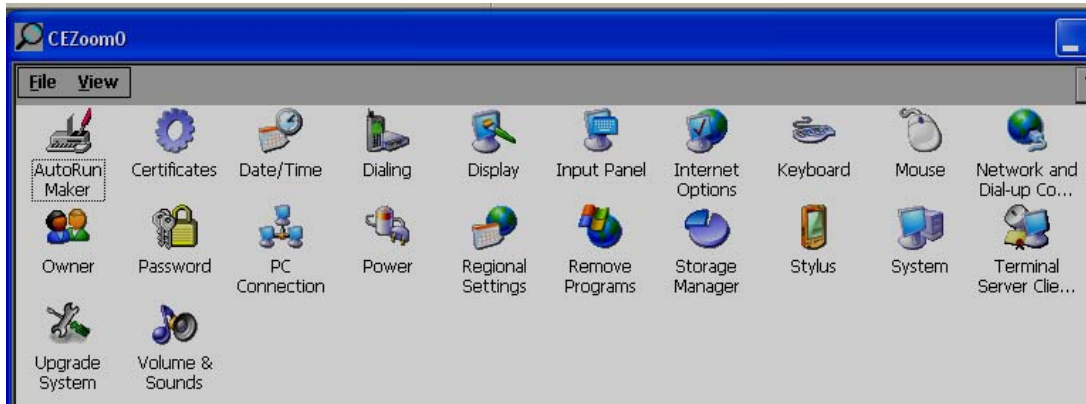


Note:

USB stick will be shown at “Hard Disk” at “My device”

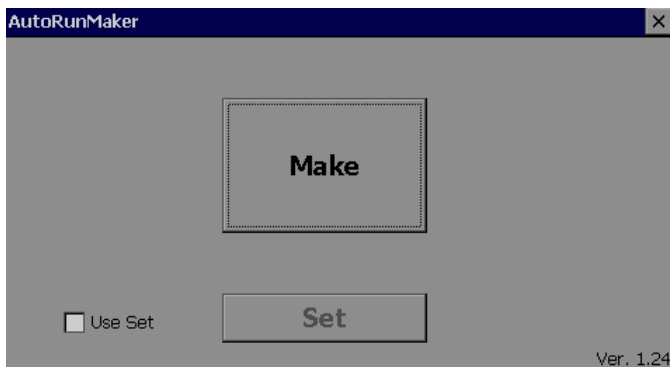
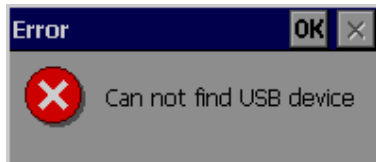
Onboard flash memory will be shown as “Resident Flash” at “My device”

Select “Control Panel” and open “Control Panel”



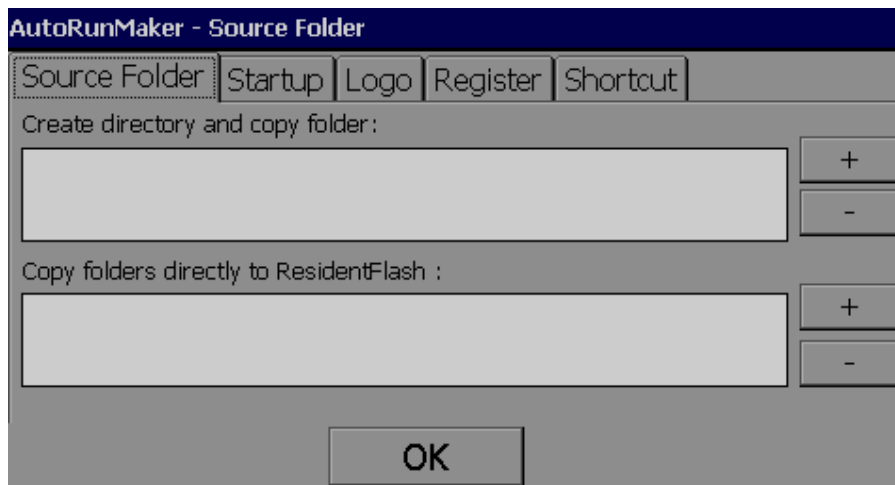
Select “AutoRun Maker” and open “AutoRun Maker”

If there is no USB device, it shows following message

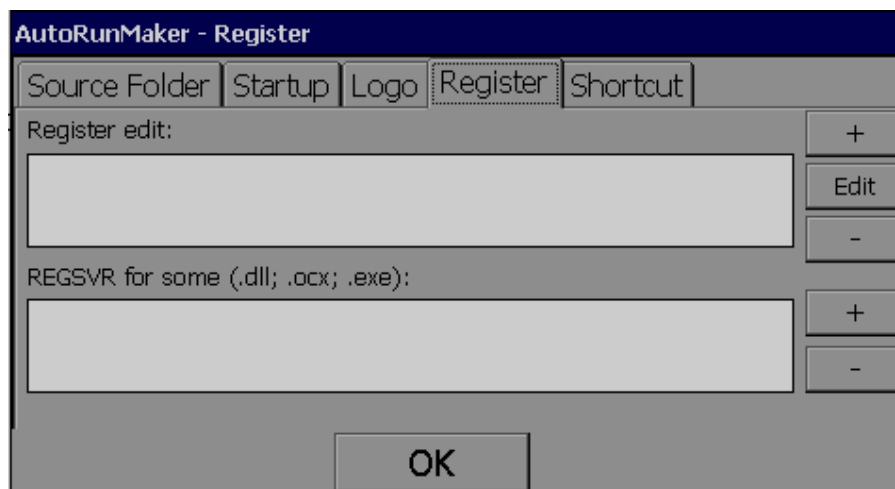


Note: Bottom Right corner shows current version of WinCE image

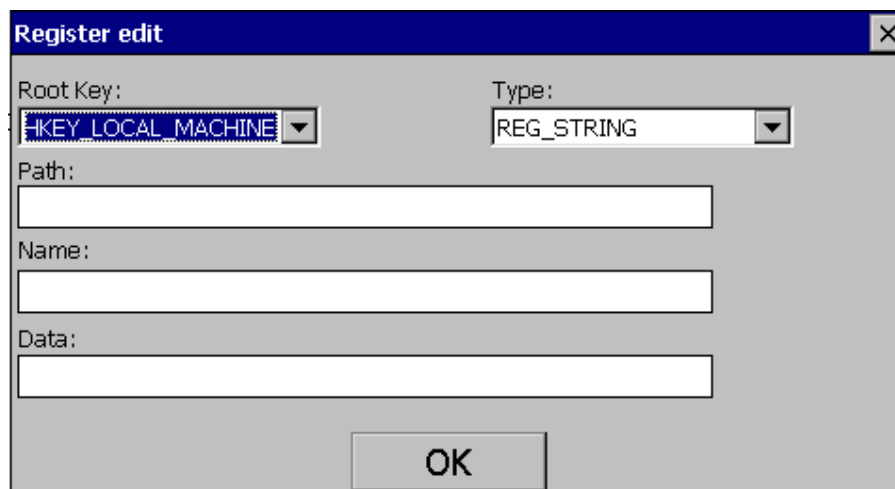
Select “Use Set” check box and press “Set”




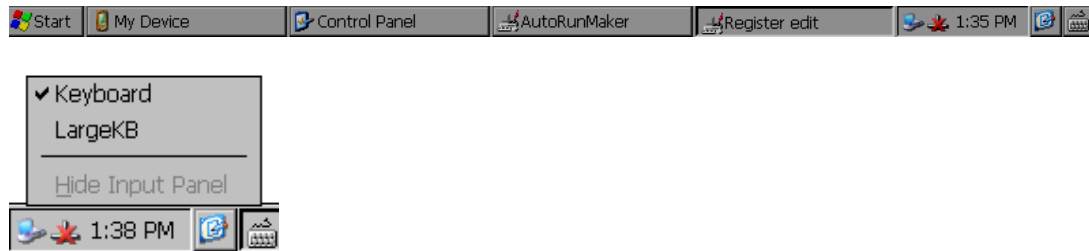
Select 'Register' Tab



Press 



If you are aware of Registry files, you can enter Path, Name and Data. To enter alphanumeric text, watch out for icon  at bottom area of the screen task bar



Press at "Keyboard"



Press at LargeKB to appear larger keyboard

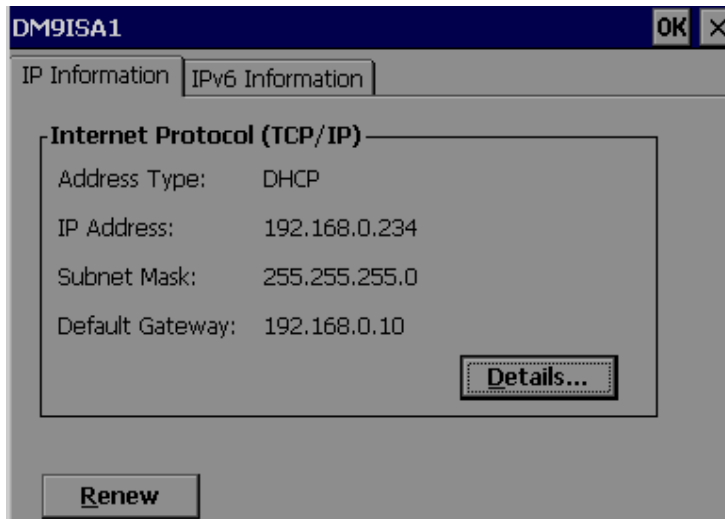


Example:

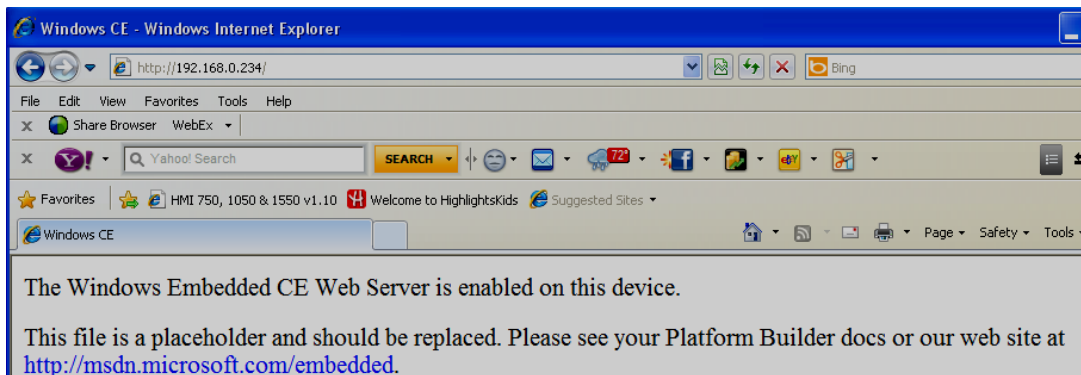
How to Configure Web Root Directory for the Web Server

By default, Web Server path is set to following location





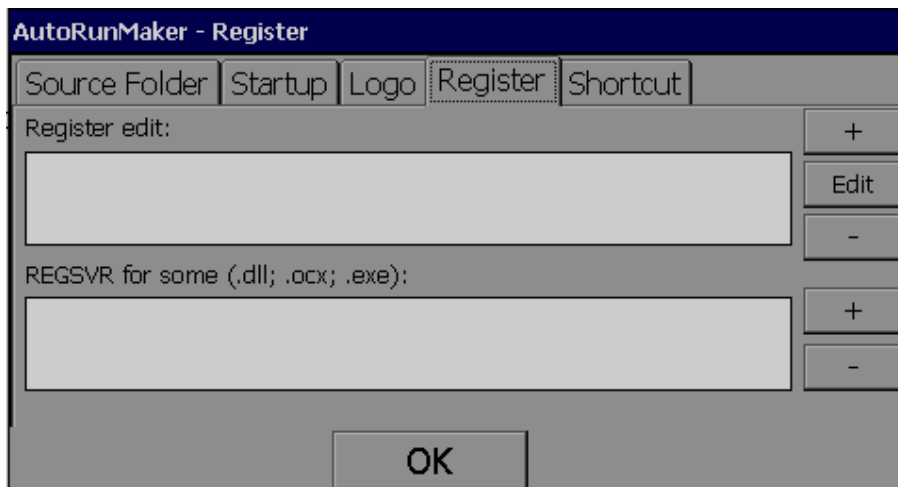
If you try to access default.html from PC, it show following message



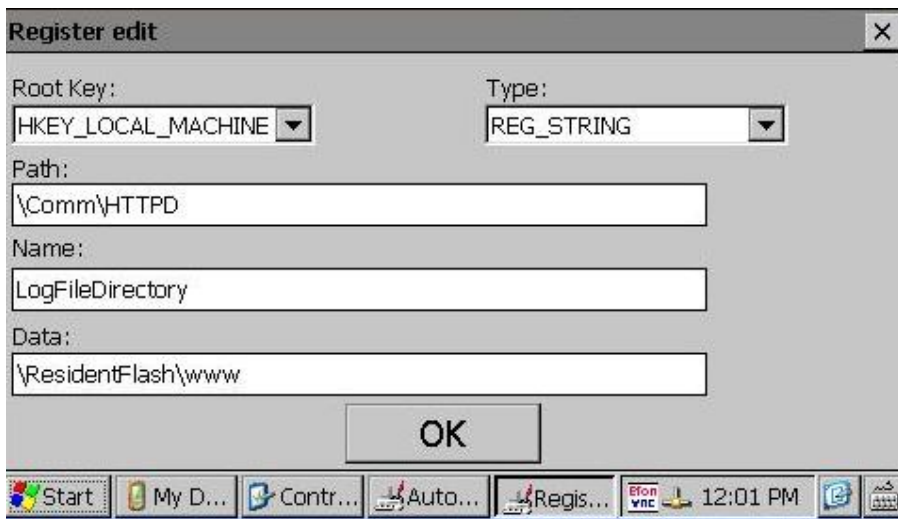
Task: How to change path for the Web server

New Target Path: \ResidentFlash\www

Step 1: Open Auto Run Maker and press "Register" Tab



Press 

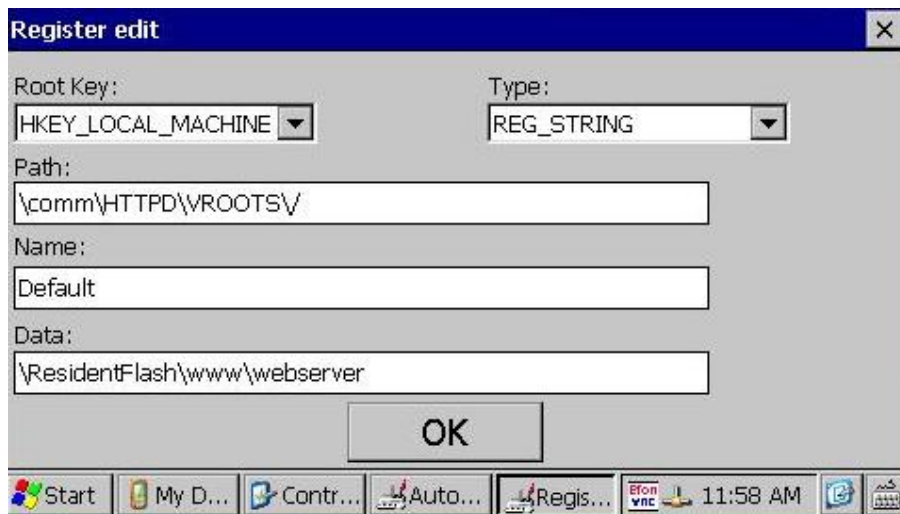


"\Comm\HTTPD\" This is original default path for the Web Server

LogFileDirectory: Original Directory Name

"\ResidentFlash\www" This is new root path for the Web Server

Enter above details using Keypad and then press "OK"

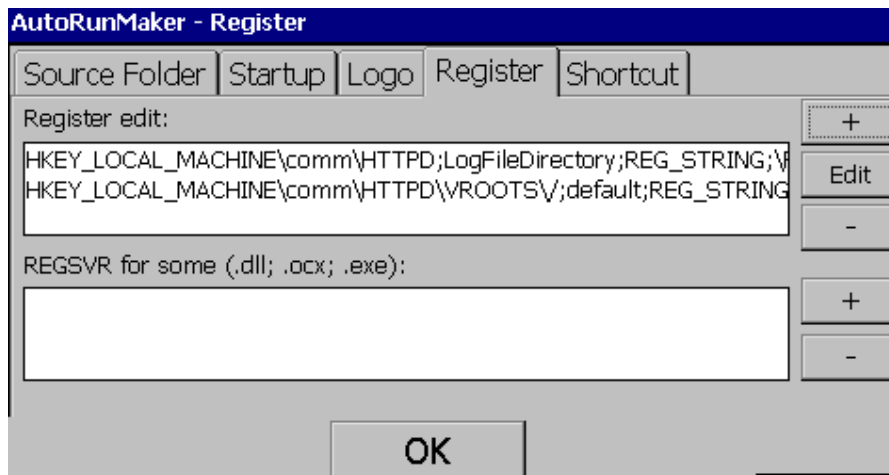


"\Comm\HTTPD\ROOTSV" This is default path for Home Page folder

(Note: Please observe the above name "ROOTSV" Carefully. The last character is not English character "V", instead it is one backward slash and one forward slash \ / but, while doing data entry, make sure, there is no gap between backward slash and forward slash and appears as English character "V")

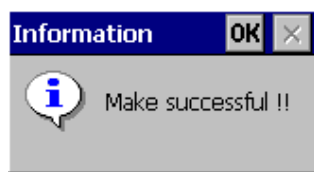
"\ResidentFlash\www\webserver" is new path for Home page folder

Enter details as shown above and press "OK"




Press "OK"

Press "Make"



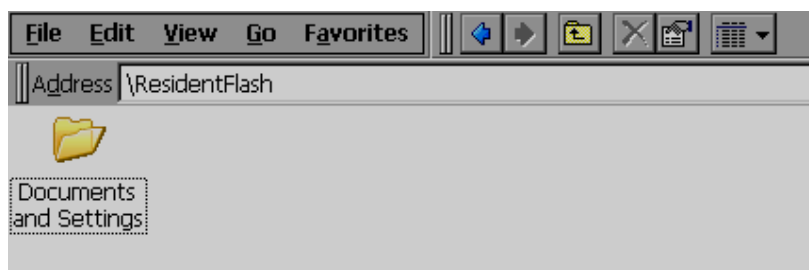
Press OK” and close “Auto run maker”

If keypad is still appearing, you can hide it from the icon  from bottom right corner

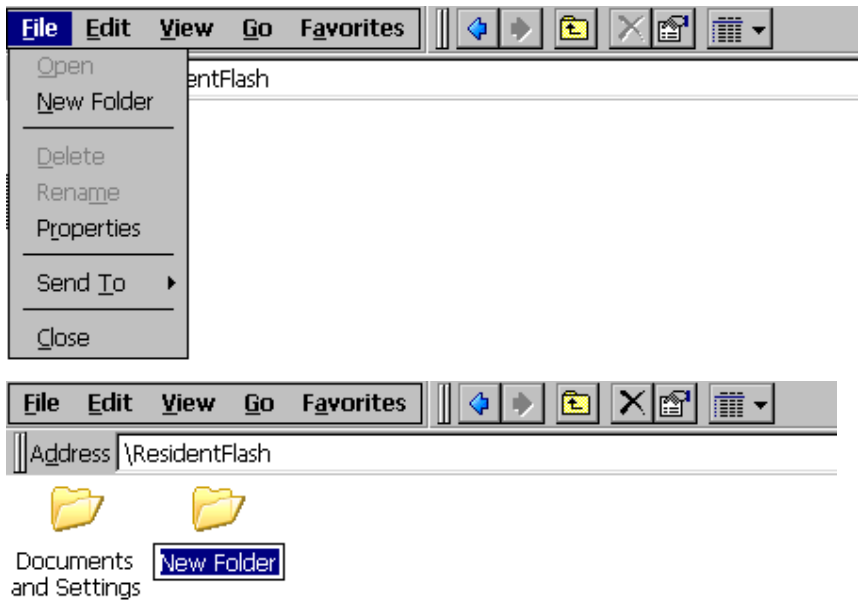
Step 3:

Open “My Device”

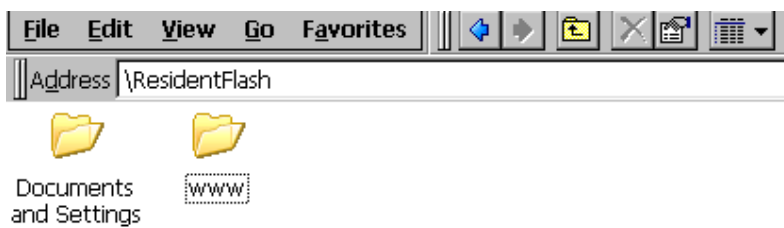
Open “ResidentFlash”



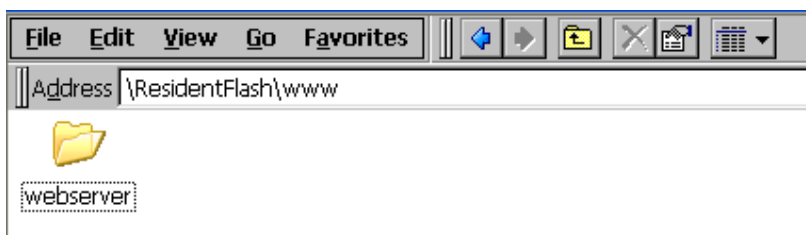
Create a new folder www at Resident Flash



Open Keypad from bottom right corner and enter name www



Create a new folder webserver at www. The procedure is same as explained above



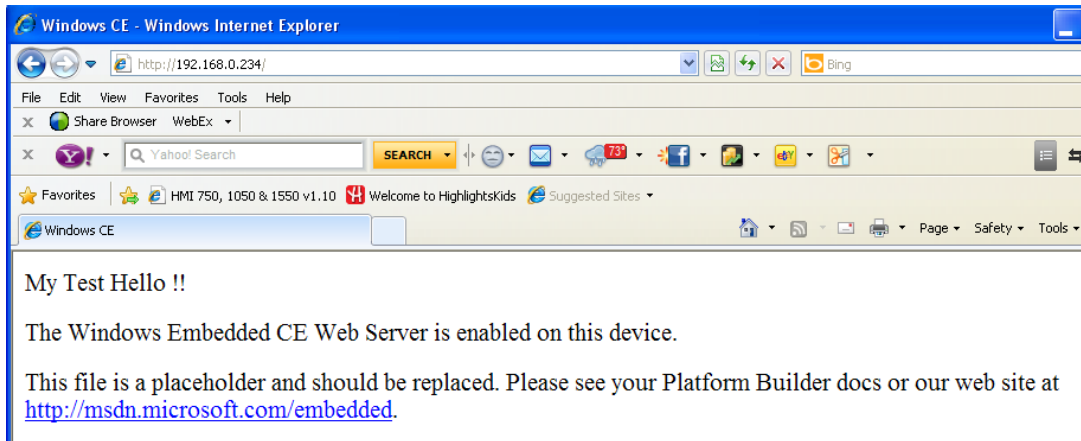
Step 4: Now, copy your customized default home page to the Resident flash at the following path

\ResidentFlash\www\webserver



Step 5: Switch Off Power supply to HMI and Power ON again

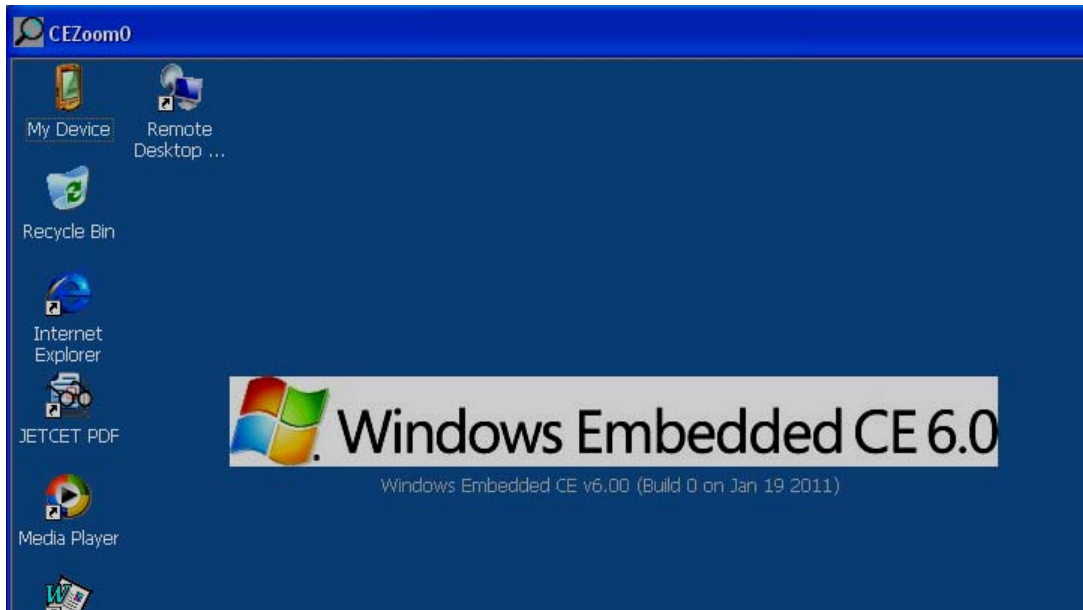
Step6: Use any browser of client to connect web server.



Note: This is modified default file loaded to \ResidentFlash\www\webserver and it is different than default html file available in WinCE image

In this HTML file, you can see a new text “My Test Hello”

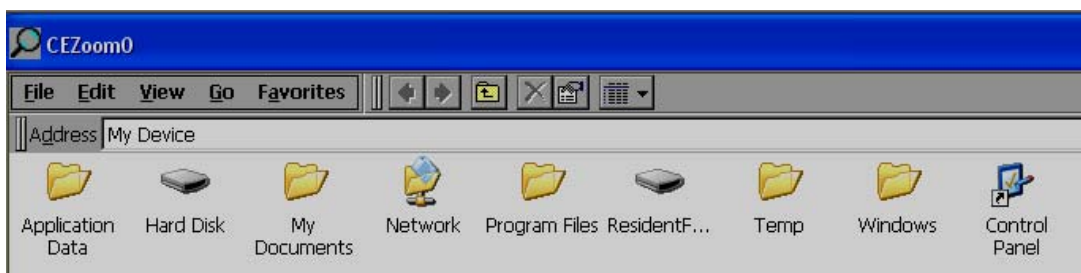
2.1.1.4 How to create shortcut at Desktop & Program



Press “My Device”, first, it will be selected and it shows with blue color mark as shown below



Now, press “My Device” quickly two times to open this folder

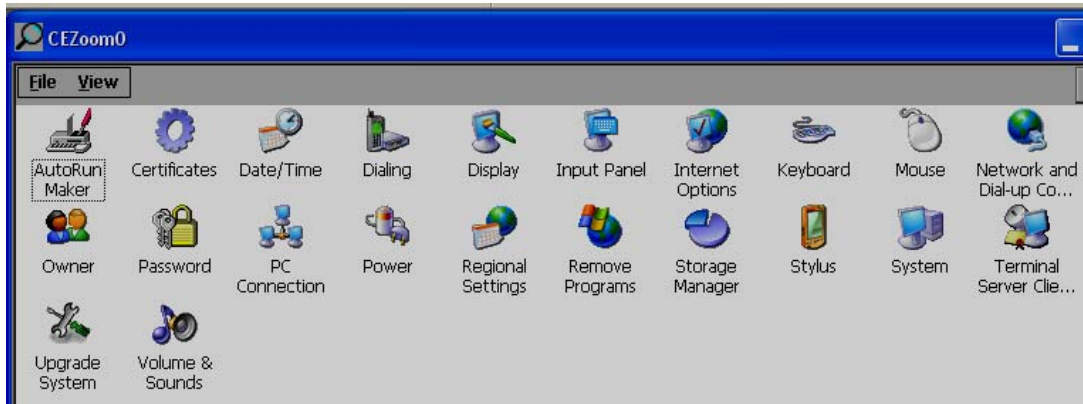


Note:

USB stick will be shown at “Hard Disk” at “My device”

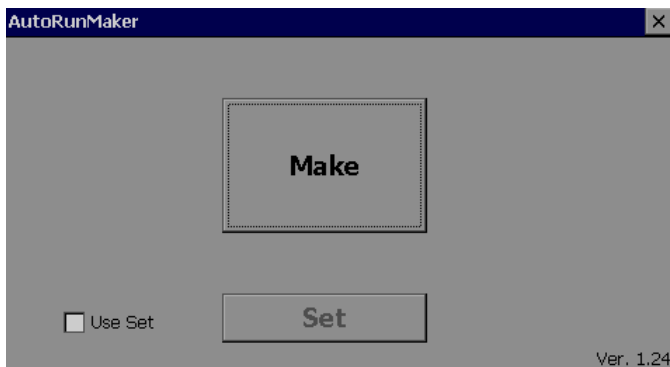
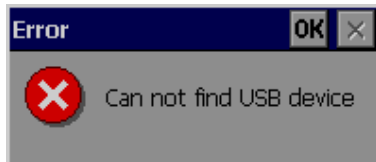
Onboard flash memory will be shown as “Resident Flash” at “My device”

Select “Control Panel” and open “Control Panel”



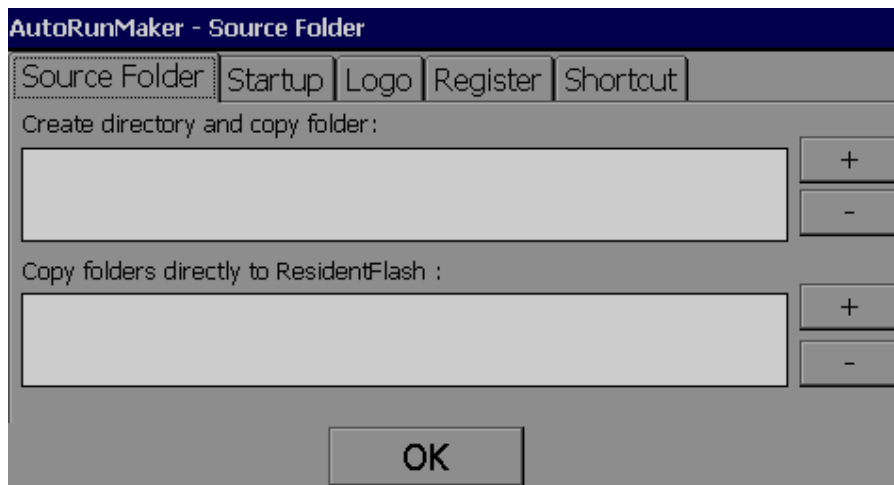
Select "AutoRun Maker" and open "AutoRun Maker"

If there is no USB device, it shows following message

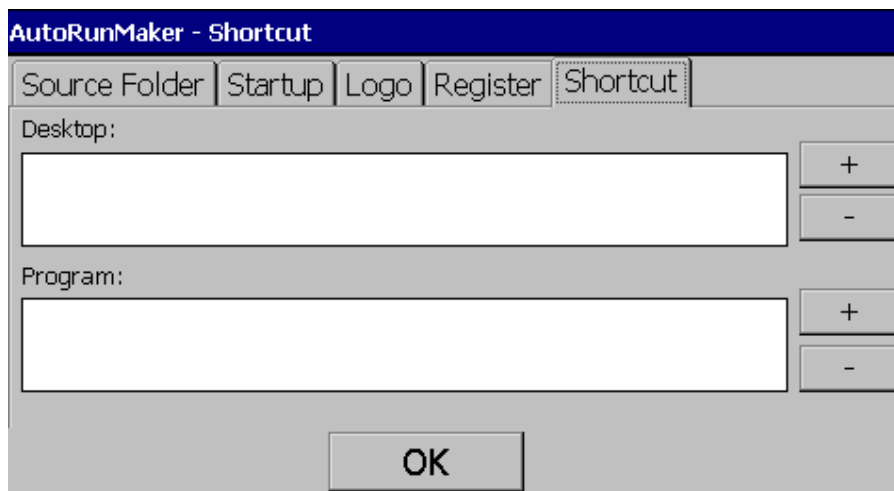


Note: Bottom Right corner shows current version of WinCE image

Select "Use Set" check box and press "Set"



Press "Shortcut" Tab





Desktop: It is to select shortcut at desktop

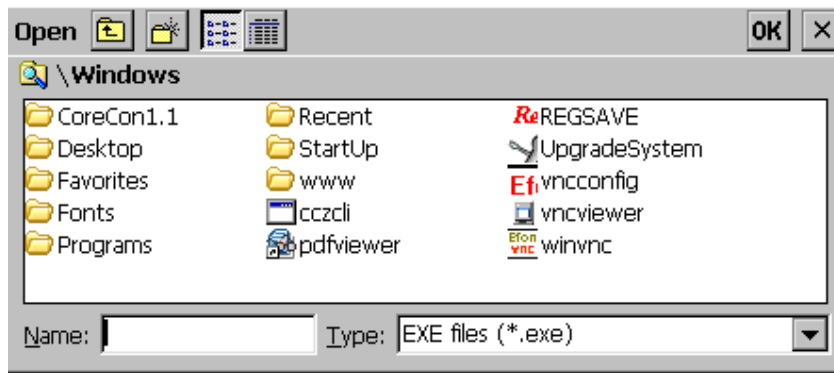
Program: It is to select shortcut at Program. Then, you can access this from Start at bottom-left corner

Example

Task: Create desktop shortcut for the following applications available at Windows folders

REGSAVE
Vncconfig
Vncviewer
Winvnc

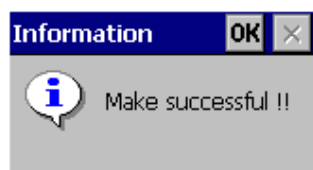
Press  and then move to Windows folders using icon 



Select the required applications

Press "OK" and follow on screen instructions

Press "Make"

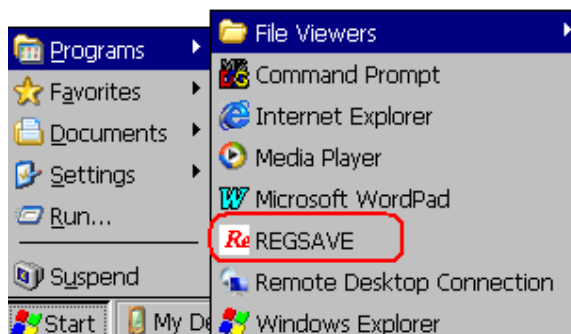
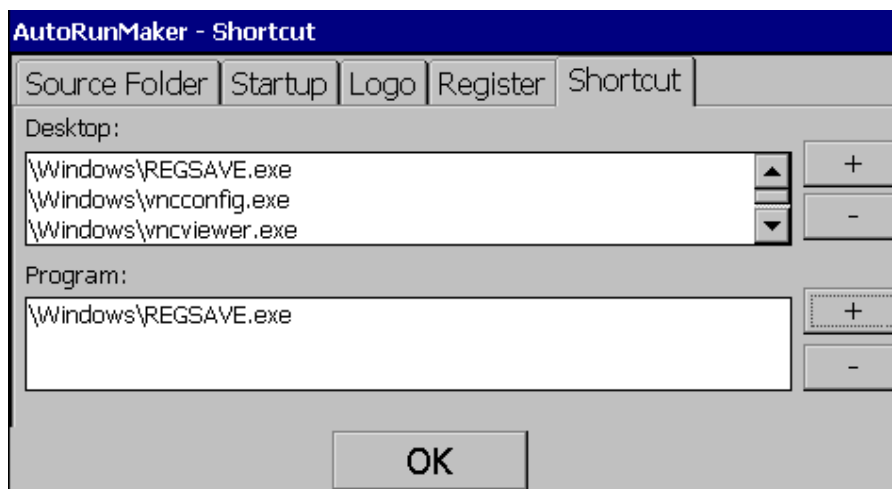


Press "OK" and close "Auto run maker"

Close all the open screens and check desktop screen



The procedure to create shortcut for program is similar to the above and once they are done, they can be accessed from the Start button as shown below

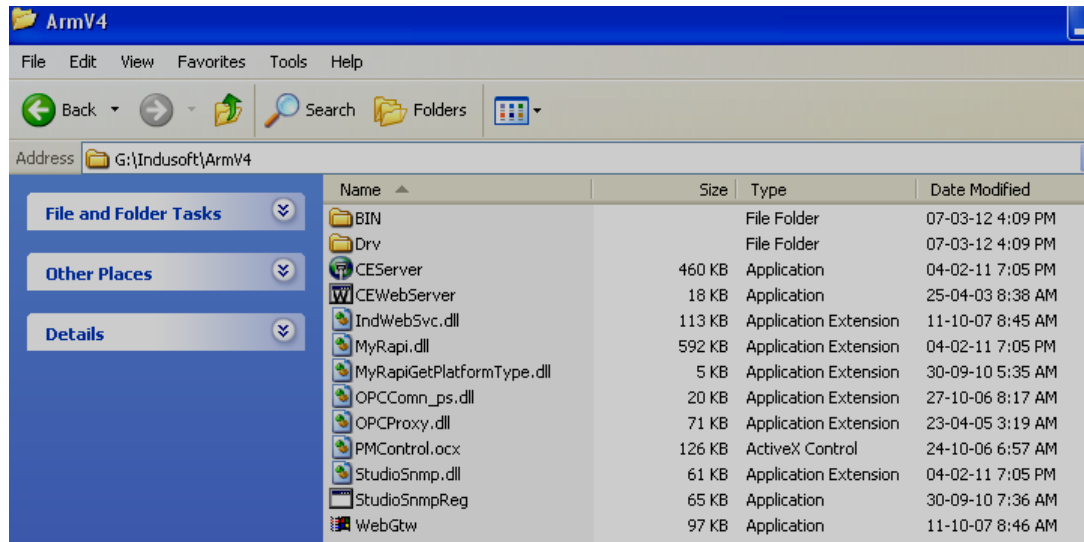


2.1.1.5 How to run *Indusoft* application in HMI

Basic Requirements: HMI loaded with Windows CE 6.0 professional

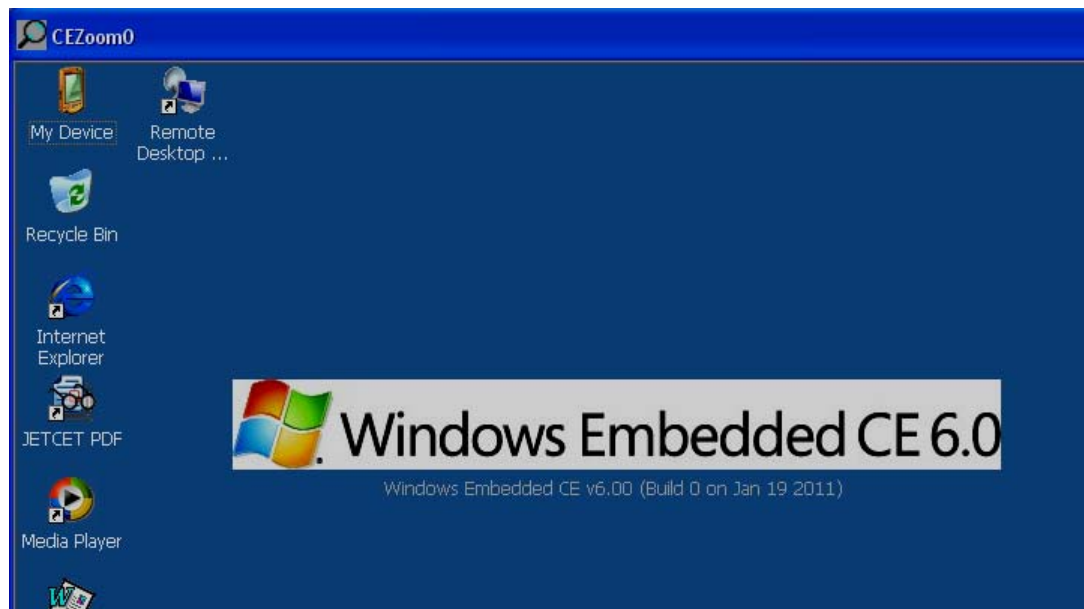
Example: How to start CEServer.EXE from Indusoft® at HMI Power ON

Step-1: Prepare a folder by name Indusoft\ArmV4 at USB disk and copy all the required files including CEServer.exe into a folder by name ArmV4. The contents will be shown as below



Note: Please contact Indusoft to get all the required files to run CEServer

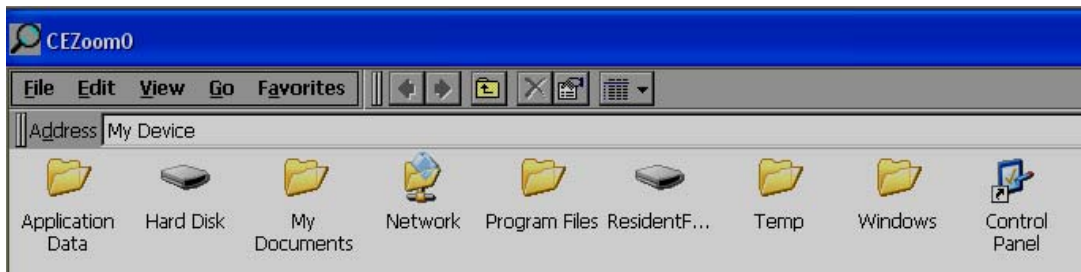
Step-2: Insert USB disk in HMI and Power ON HMI



Step-3: Press “My Device”, first, it will be selected and it shows with blue color mark as shown below



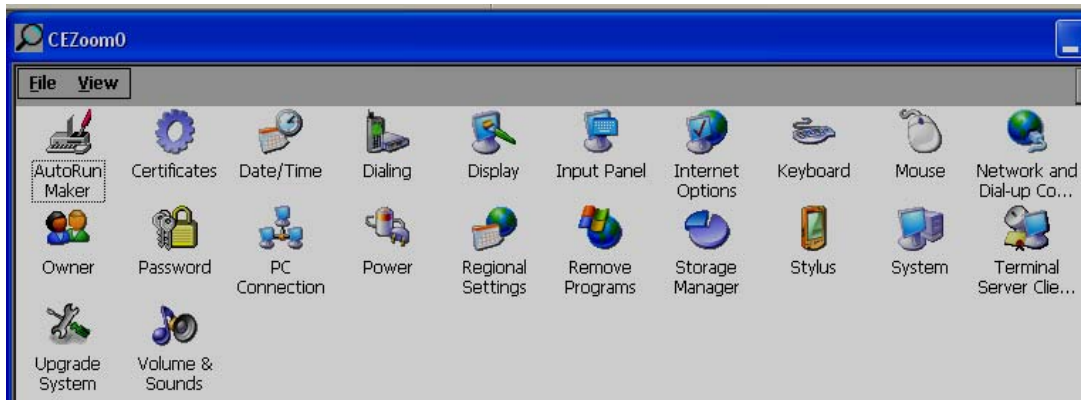
Now, press “My Device” quickly two times to open this folder



Note:

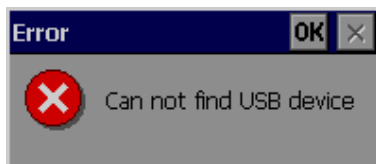
USB stick will be shown at “Hard Disk” at “My device”
Onboard flash memory will be shown as “Resident Flash” at “My device”

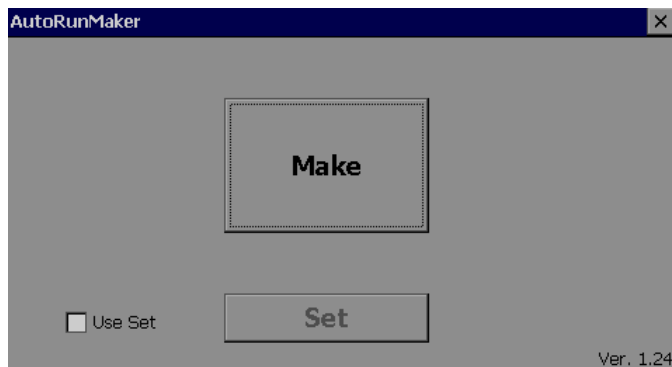
Select “Control Panel” and open “Control Panel”



Select “AutoRun Maker” and open “AutoRun Maker”

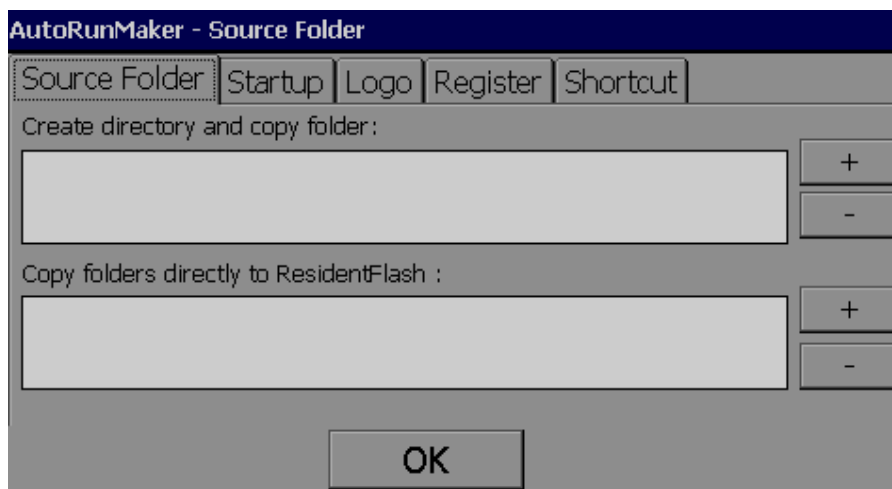
If there is no USB device, it shows following message






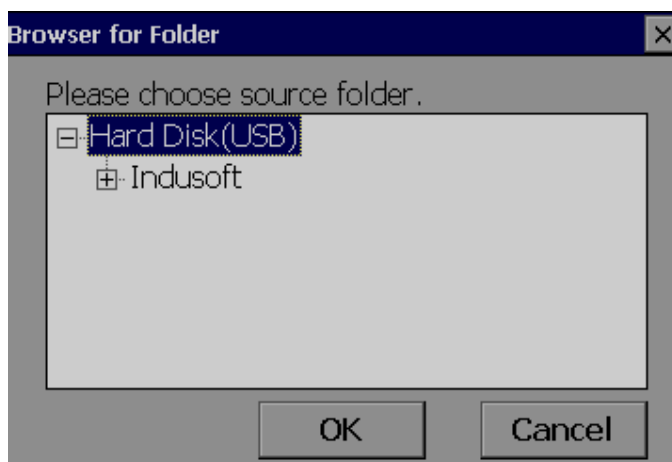
Note: Bottom Right corner shows current version of WinCE image

Select “Use Set” check box and press “Set”

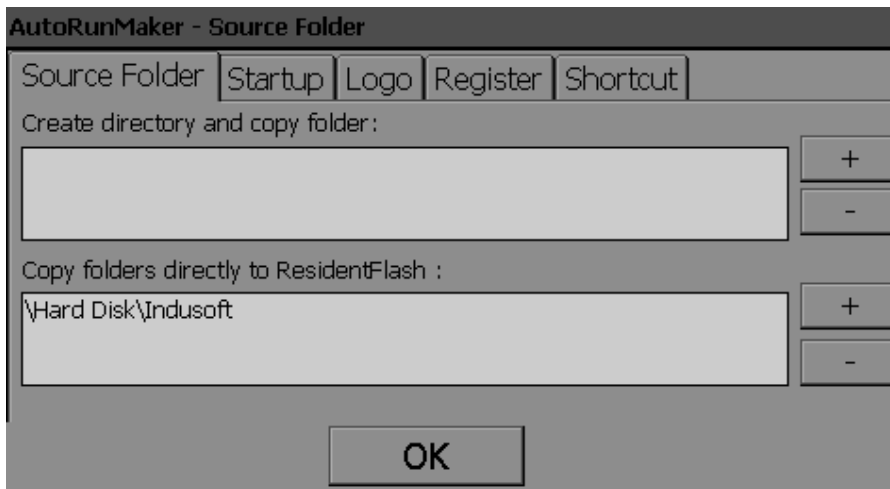
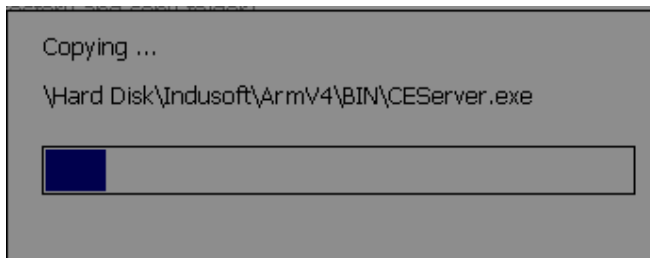


For Indusoft example, we will use “Copy folders directly to Resident flash” option in this manual

Press  button at “Copy folders directly to Resident flash”

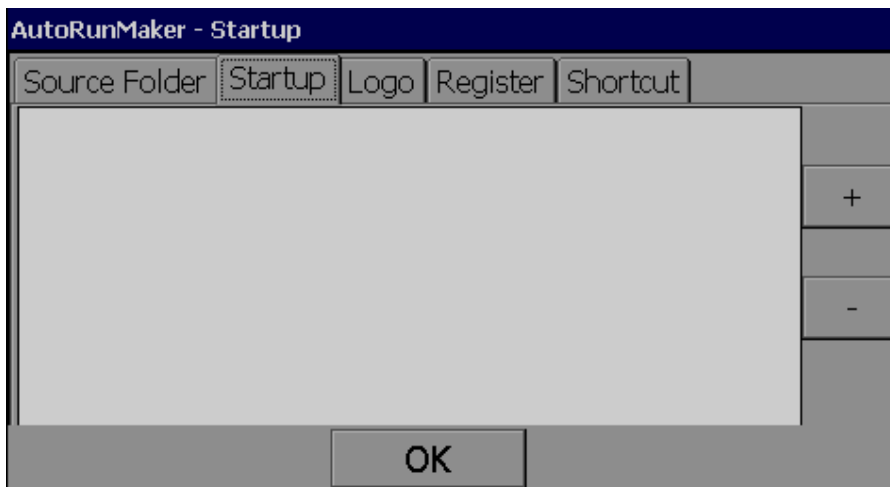


Select “Indusoft” and press “OK”

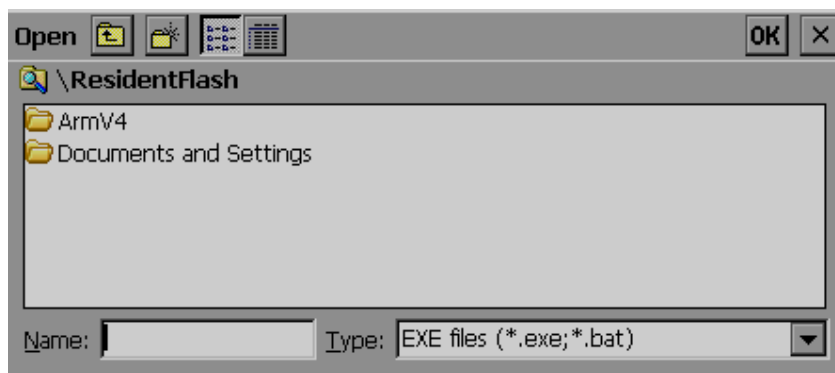


All the content from Indusoft\\Armv4 folder at USB disk is now copied to Resident flash and now, it appears the source path as shown above.

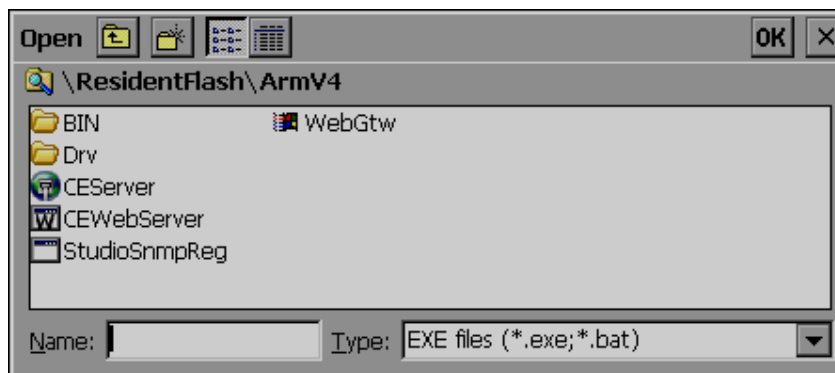
Now, press "StartUp" Tab



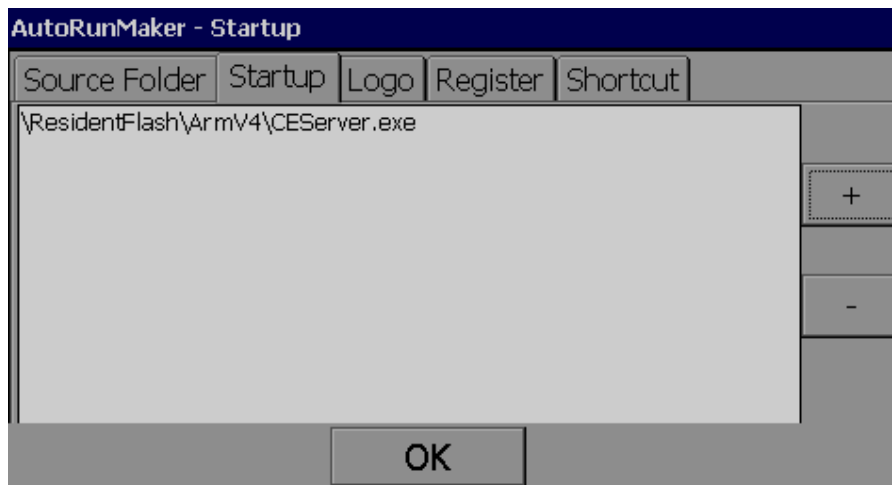
Press  button to select the path for the exe file



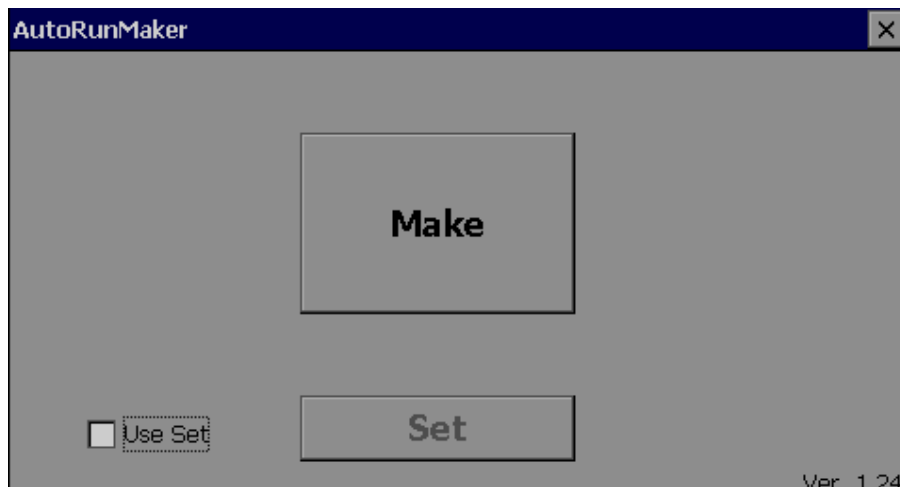
Select ArmV4 and open it



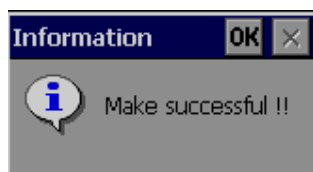
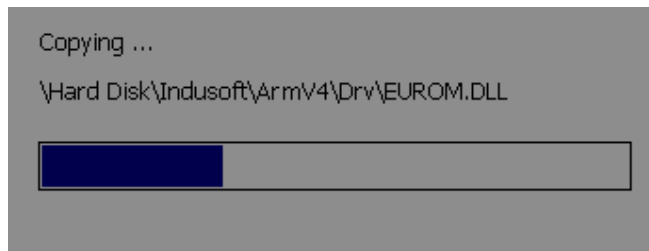
Select "CEServer.exe" and then press "OK" at top right corner



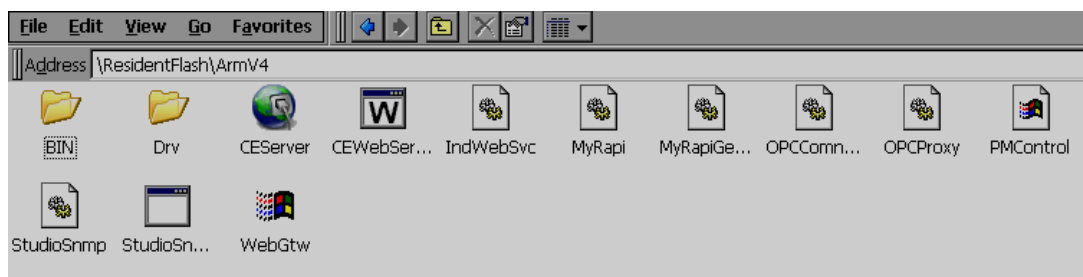
Press "OK"



Press "Make" button



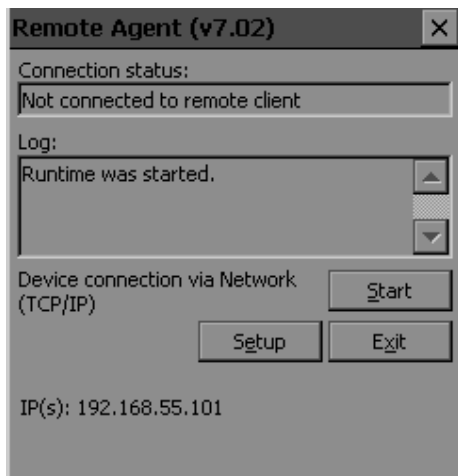
Open "MyDevice\\ResidentFlash\\ArmV4" and double check the contents



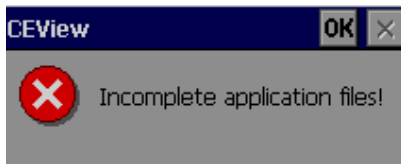
Now, remove the USB disk

Switch OFF Power supply to HMI

Power ON HMI again



CE Server start automatically at HMI Power ON as shown above



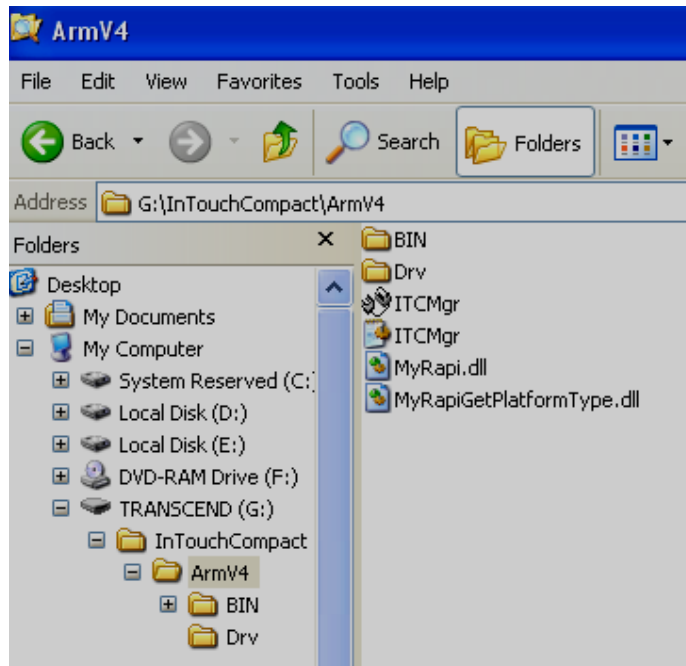
If there is any error message as shown above, you may need to download system files from PC to HMI (From Indusoft software), then, download required demo applications

2.1.1.6 How to run *Wonderware* application in HMI

Basic Requirements: HMI loaded with Windows CE 6.0 professional

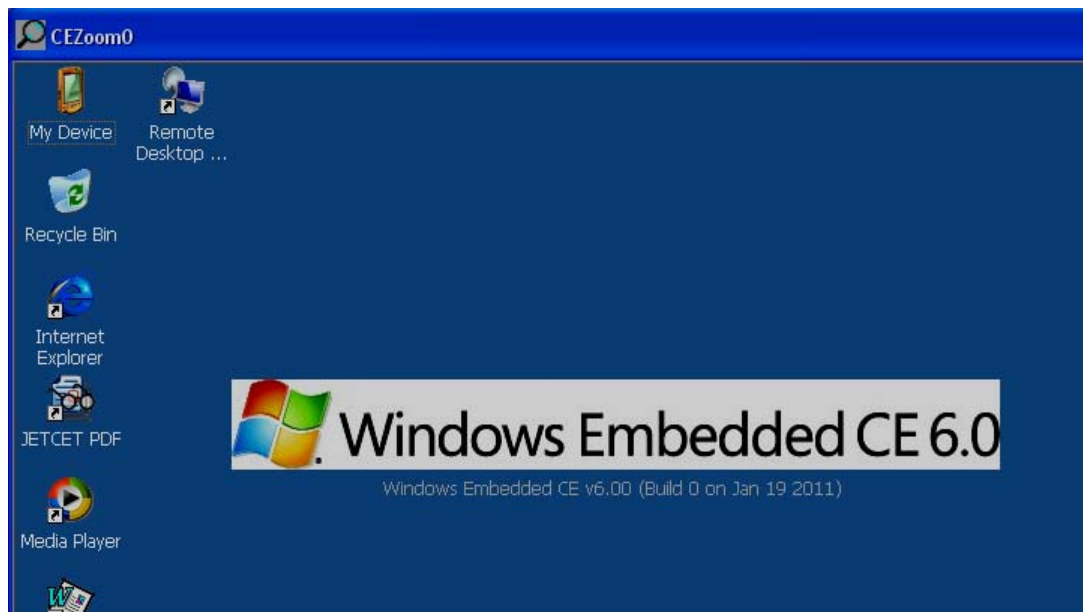
Example: How to start ITCMgr.EXE from Wonderware® at HMI Power ON

Step-1: Prepare a folder by name InTouchCompact\ArmV4 at USB disk and copy all the required files including ITCMgr.exe into a folder by name ArmV4. The contents will be shown as below



Note: Please contact Wonderware to get all the latest required files to run ITCMgr.exe

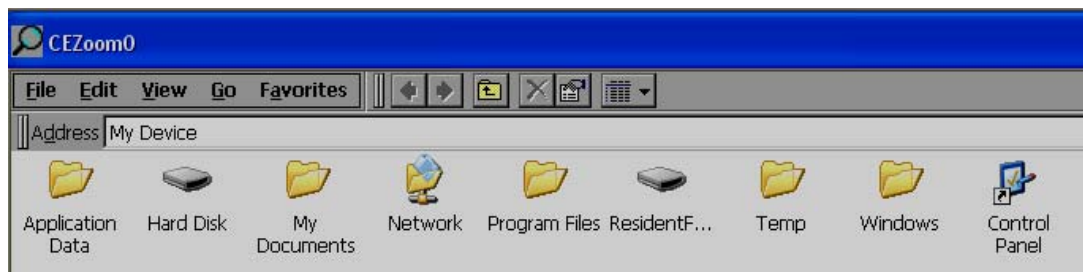
Step-2: Insert USB disk in HMI and Power ON HMI



Step-3: Press “My Device”, first, it will be selected and it shows with blue color mark as shown below



Now, press “My Device” quickly two times to open this folder

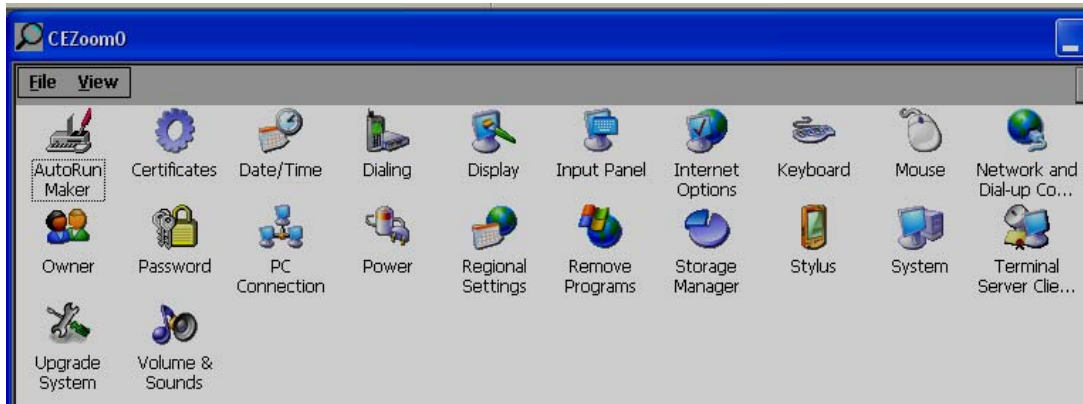


Note:

USB stick will be shown at “Hard Disk” at “My device”

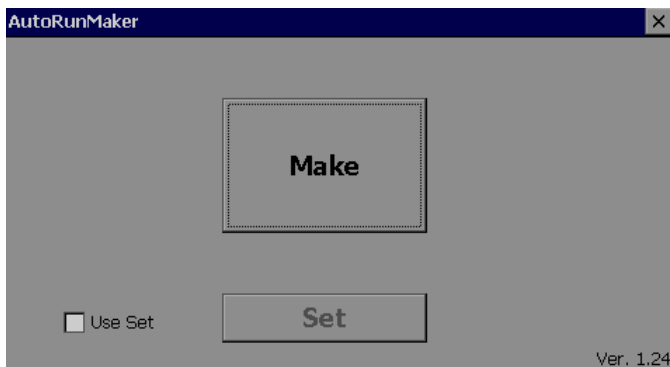
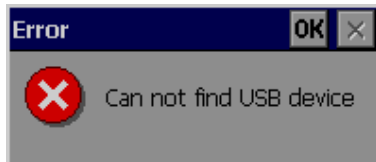
Onboard flash memory will be shown as “Resident Flash” at “My device”

Select “Control Panel” and open “Control Panel”



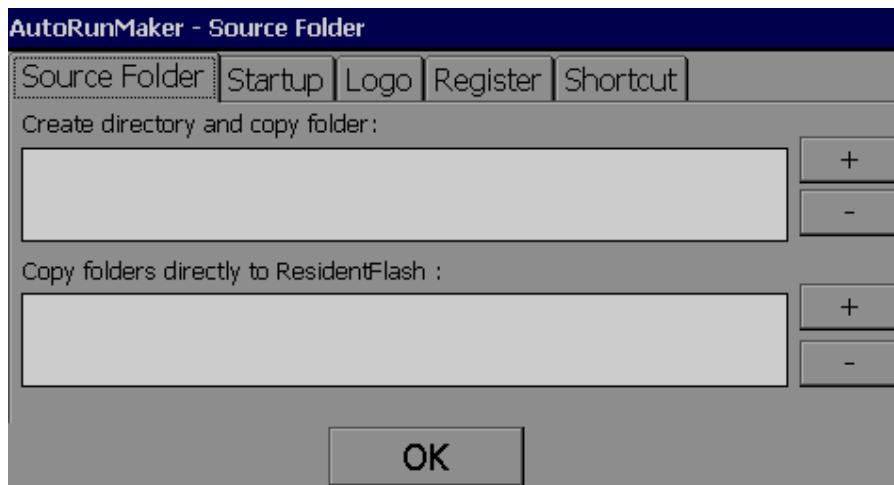
Select "AutoRun Maker" and open "AutoRun Maker"

If there is no USB device, it shows following message



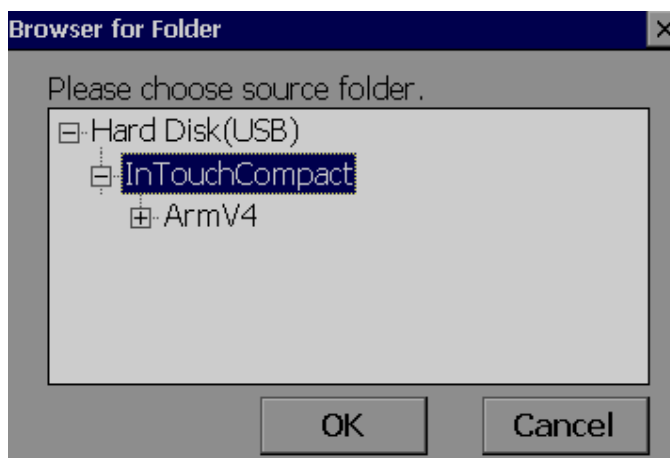
Note: Bottom Right corner shows current version of WinCE image

Select "Use Set" check box and press "Set"

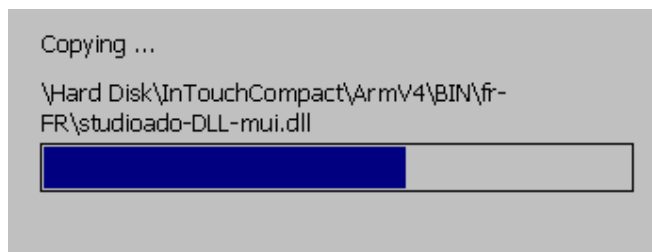


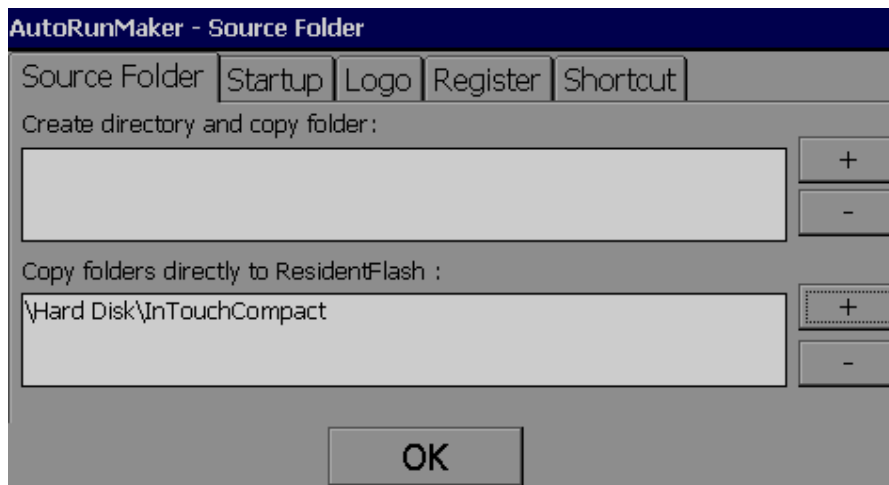
For Wonderware example, we will use “Copy folders directly to Resident flash” option in this manual

Press  button at “Copy folders directly to Resident flash”



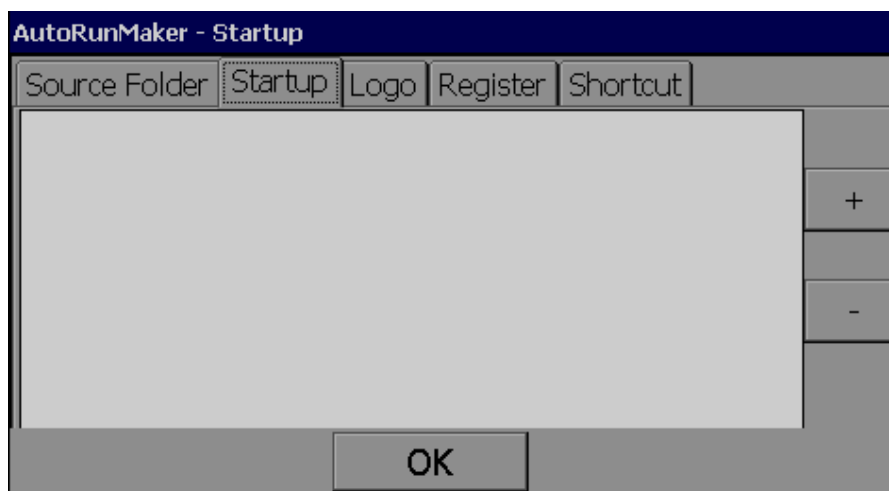
Select “InTouchCompact” and press “OK”




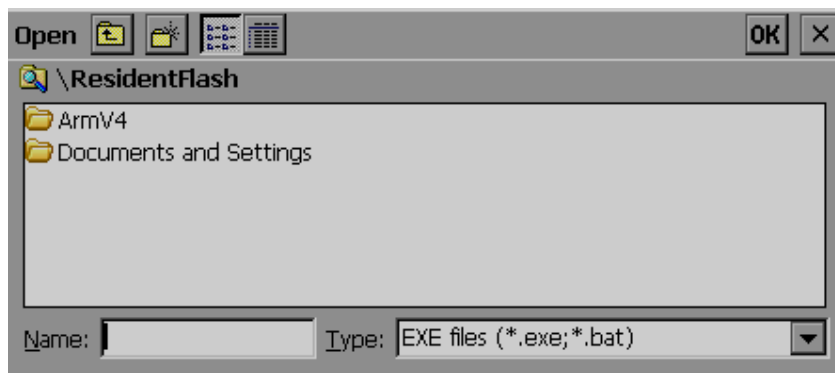


All the content from InTouchCompact folder at USB disk is now copied to Resident flash and now, it appears the source path as shown above.

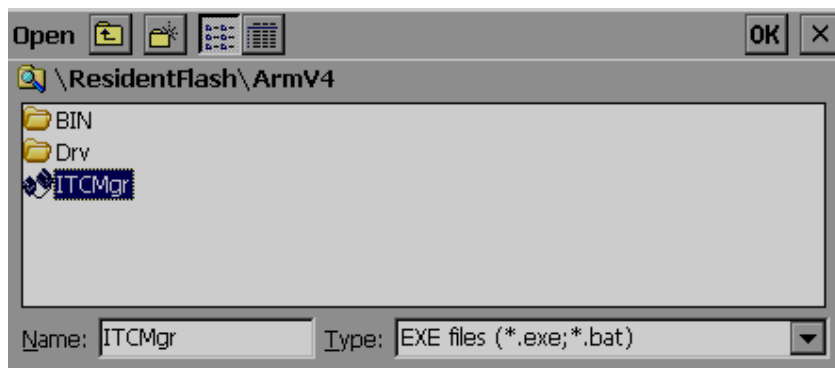
Now, press "StartUp" Tab



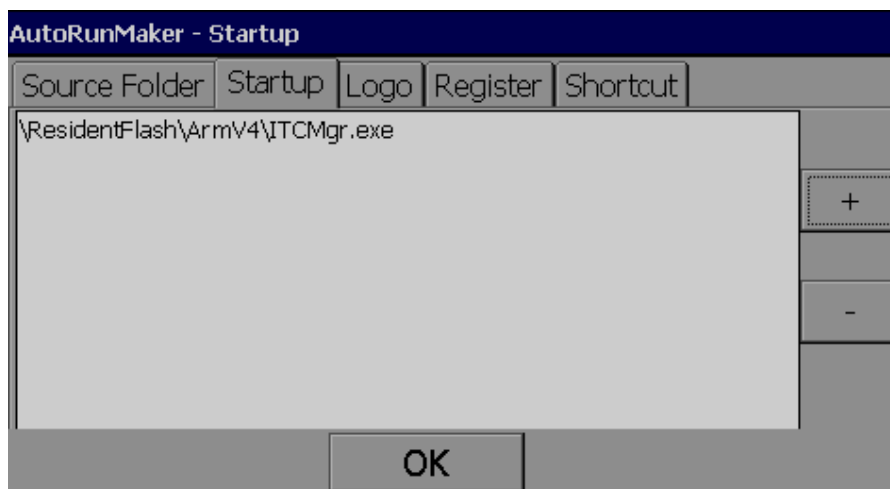
Press  button to select the path for the exe file



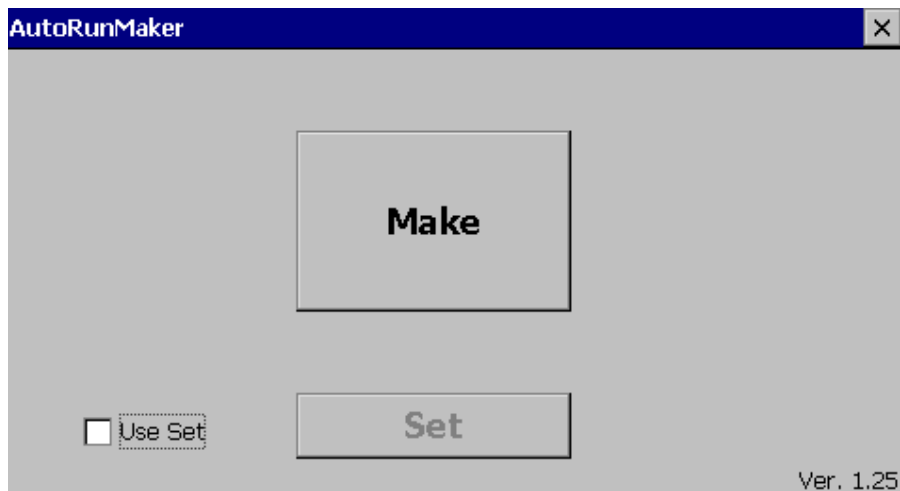
Select ArmV4 and open it



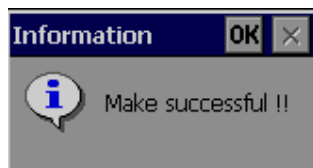
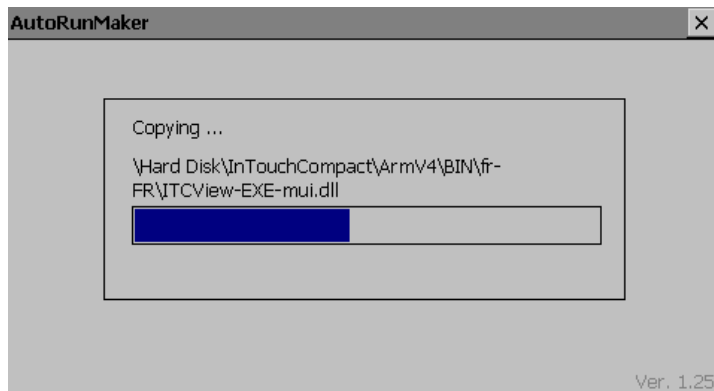
Select "ITCMgr.exe" and then press "OK" at top right corner



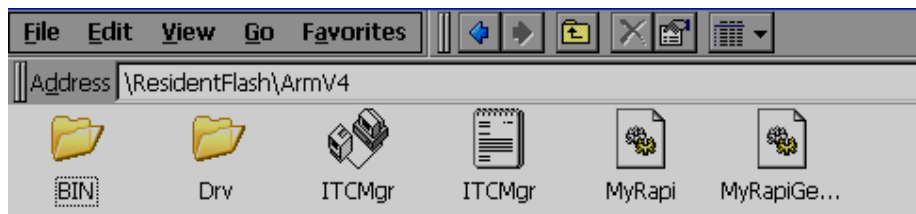
Press "OK"



Press "Make" button



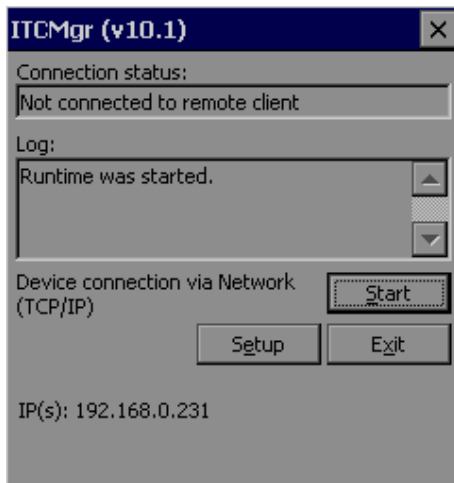
Open "MyDevice\\ResidentFlash\\ArmV4 and double check the contents



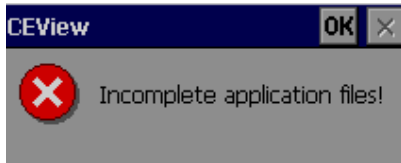
Now, remove the USB disk

Switch OFF Power supply to HMI

Power ON HMI again



ITCMgr.exe start automatically at HMI Power ON as shown above

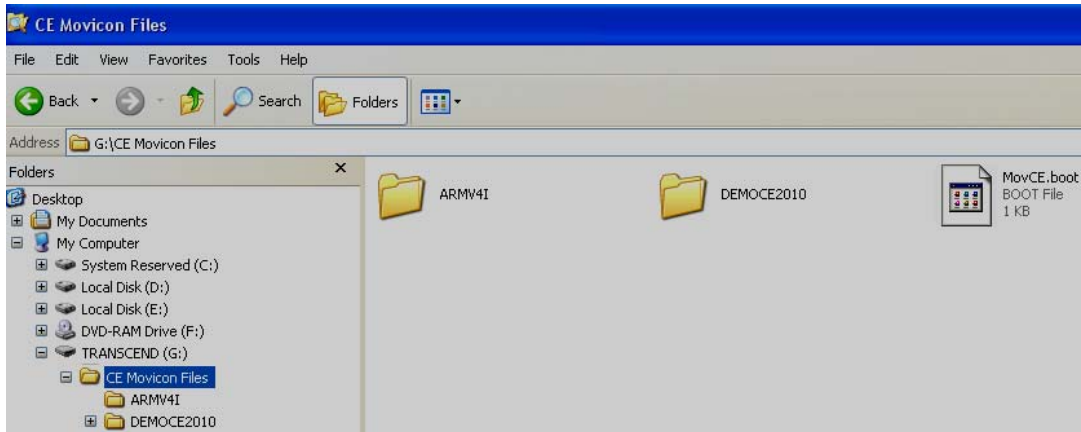


If there is any error message as shown above, you may need to download system files from PC to HMI (From Wonderware software), then, download required demo applications

2.1.1.7 How to run *Movicon* application in HMI

Basic Requirements: HMI loaded with Windows CE 6.0 professional

Step-1: Prepare a folder by name CE Movicon Files at USB disk



Note: Please contact Progea to get all the latest required files to run CEUploadServer.exe & MovCE.exe

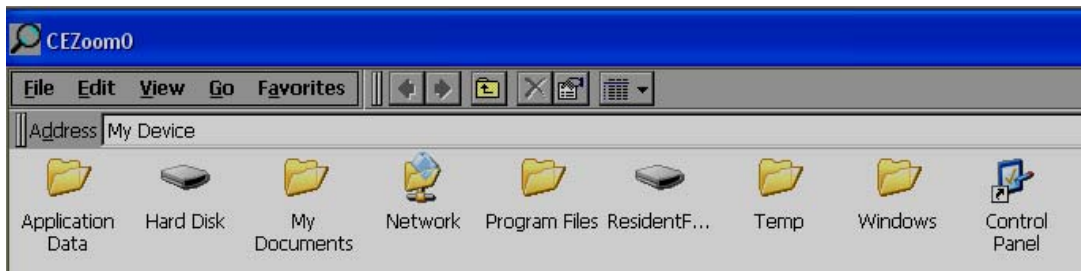
Step-2: Insert USB disk in HMI and Power ON HMI



Step-3: Press “My Device”, first, it will be selected and it shows with blue color mark as shown below



Now, press “My Device” quickly two times to open this folder

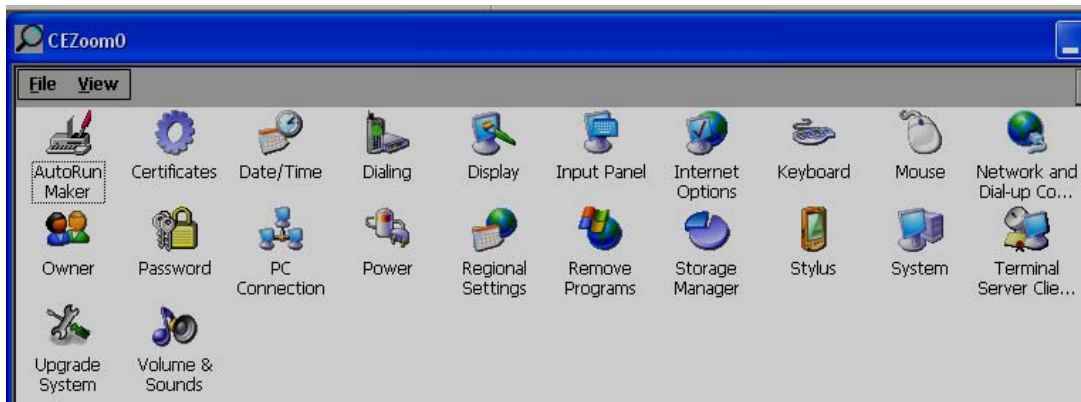


Note:

USB stick will be shown at “Hard Disk” at “My device”

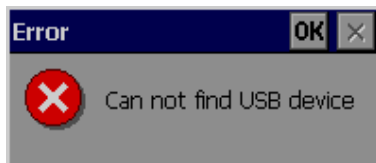
Onboard flash memory will be shown as “Resident Flash” at “My device”

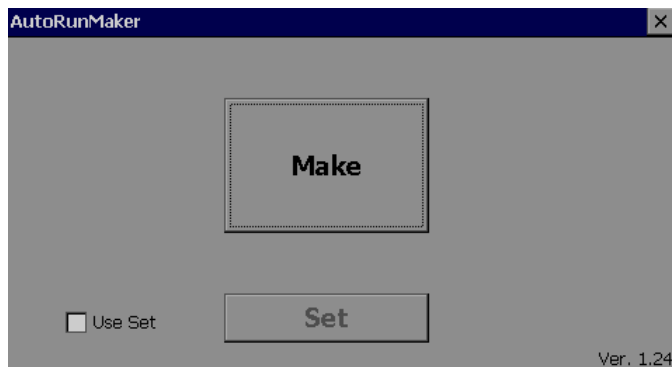
Select “Control Panel” and open “Control Panel”



Select “AutoRun Maker” and open “AutoRun Maker”

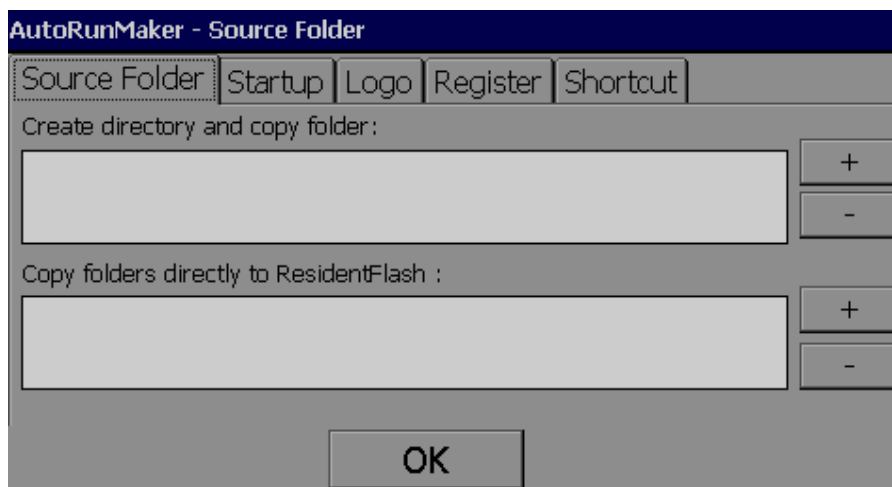
If there is no USB device, it shows following message






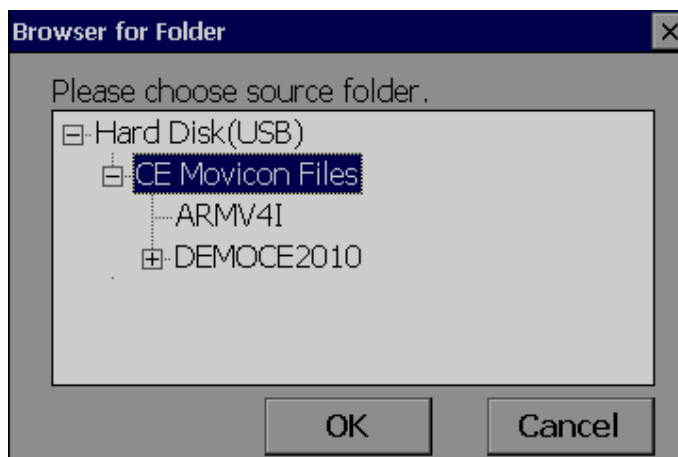
Note: Bottom Right corner shows current version of WinCE image

Select “Use Set” check box and press “Set”

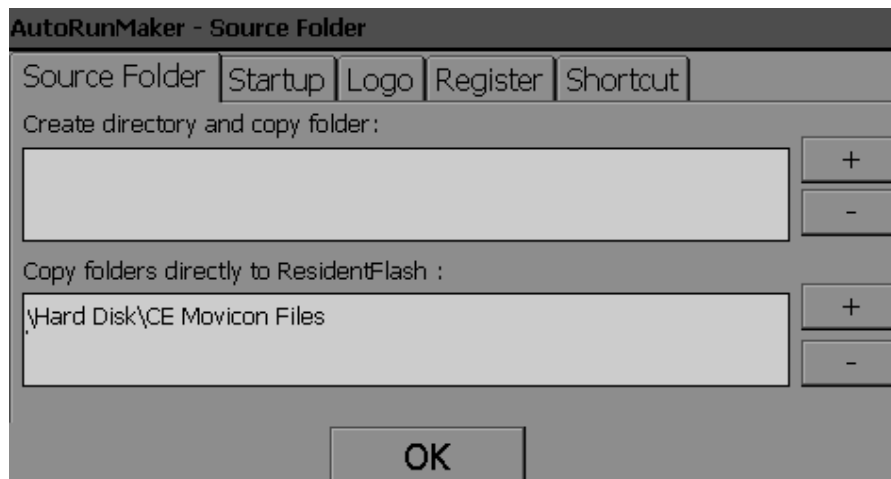
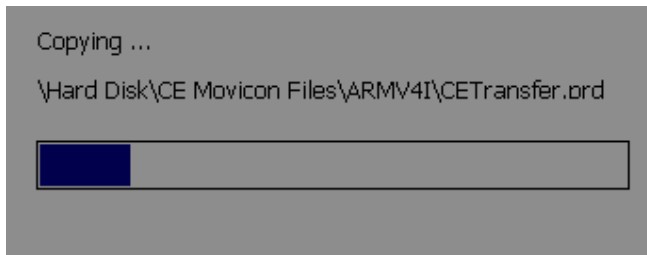


For Movicon example, we will use “Copy folders directly to Resident flash” option in this manual

Press  button at “Copy folders directly to Resident flash”

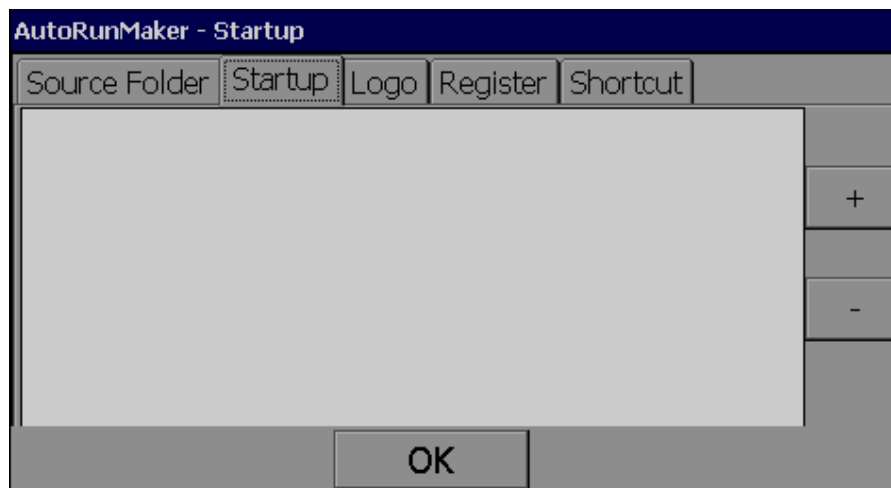


Select “CE Movicon Files” and press “OK”

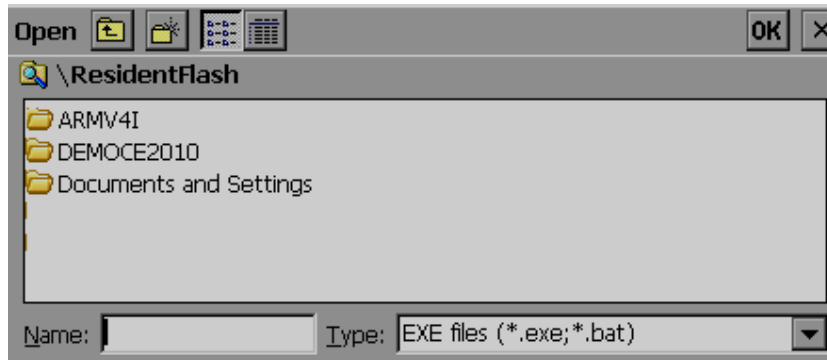


All the content from CE Movicon Files folder at USB disk is now copied to Resident flash and now, it appears the source path as shown above.

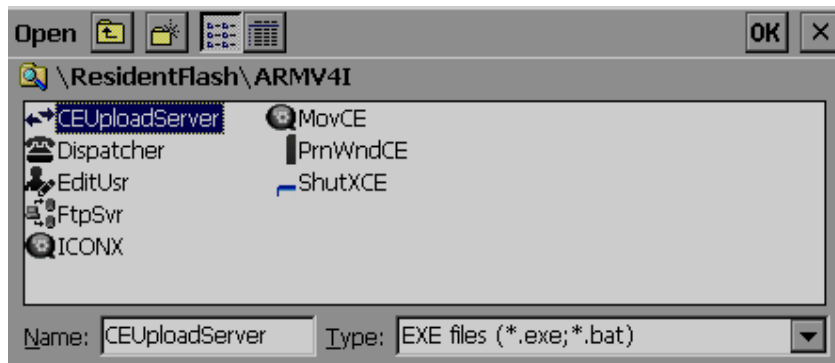
Now, press “StartUp” Tab



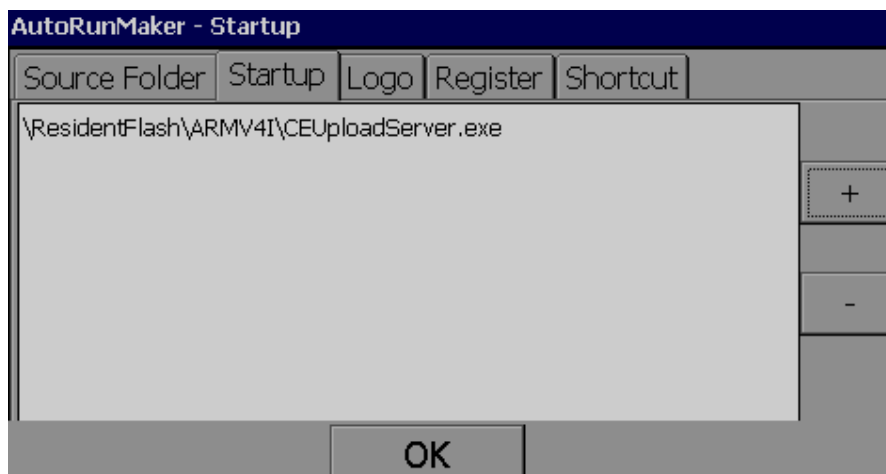
Press  button to select the path for the exe file



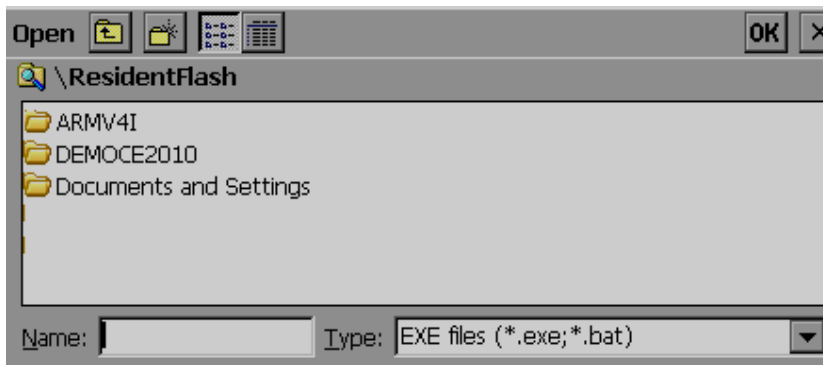
Select ArmV4I and open it



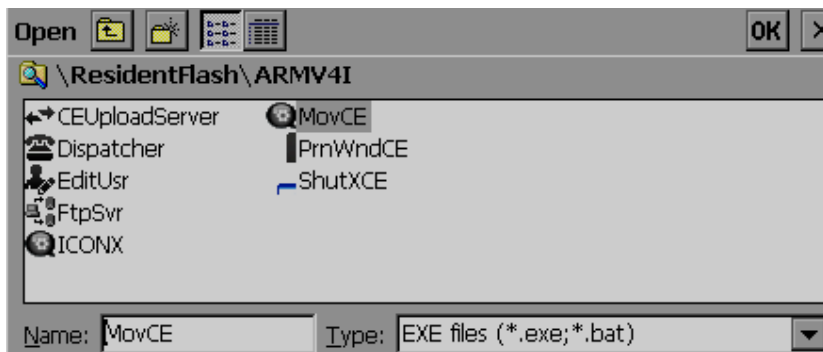
Select "CEUploadServer.exe" and then press "OK" at top right corner



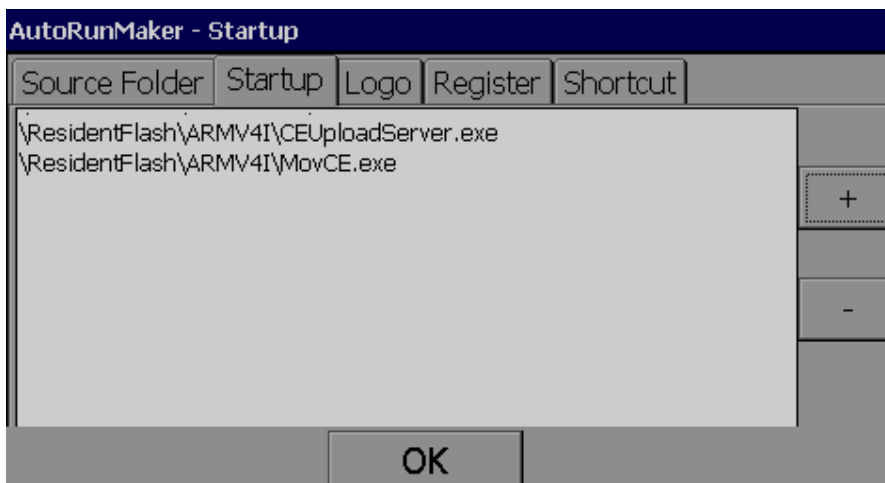
Press  button to select the path for MovCE.exe



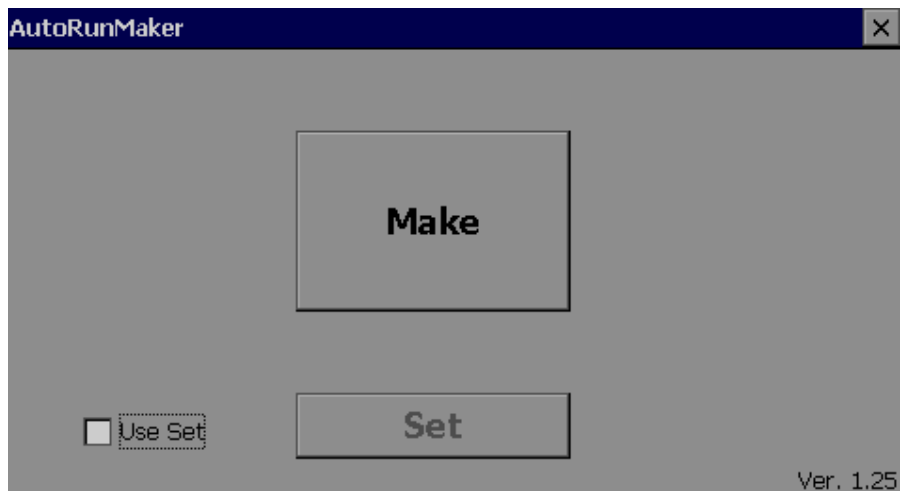
Select ArmV4I and open it



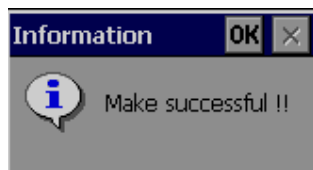
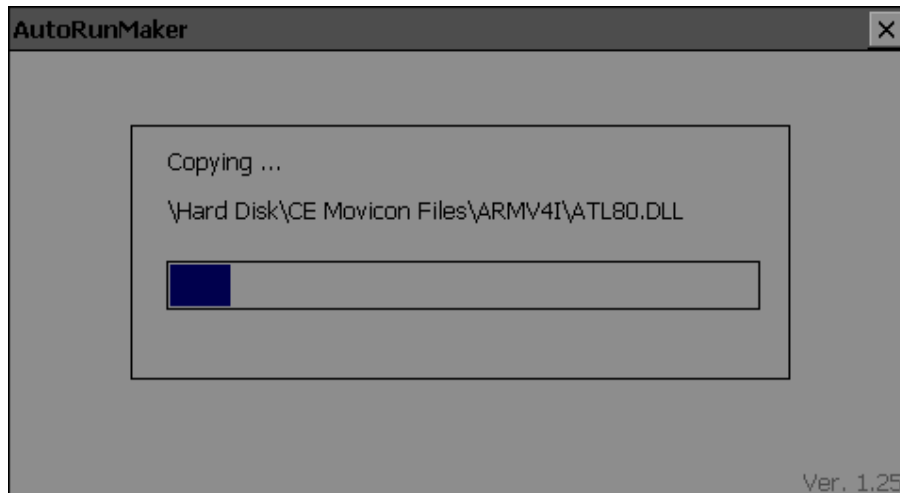
Select "MovCE.exe" and then press "OK" at top right corner



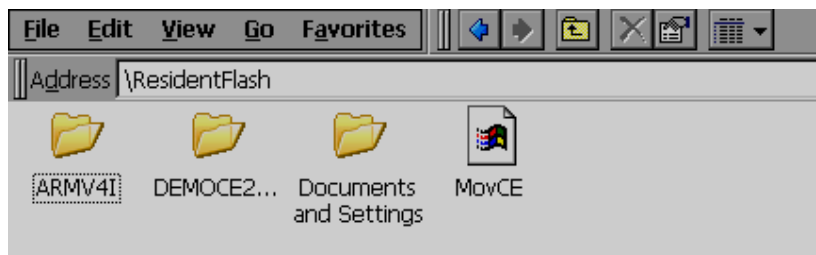
Press "OK"



Press "Make" button

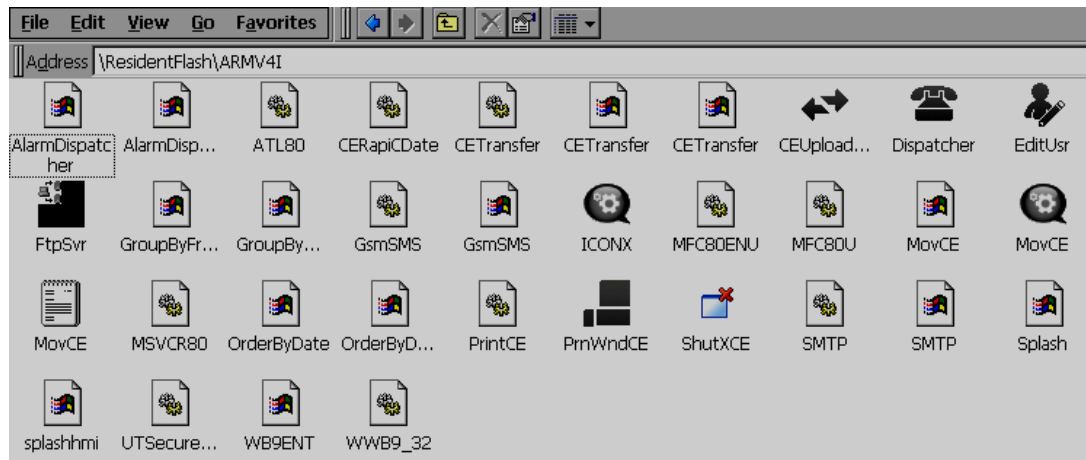


Open "MyDevice\\ResidentFlash" and check the contents



ARMV4I normally contains CEUploadServer.exe and MovCE.exe
 DemoCE2010 contains the demo project
 MovCE.boot file is required to start demo project automatically after running MovCE.exe

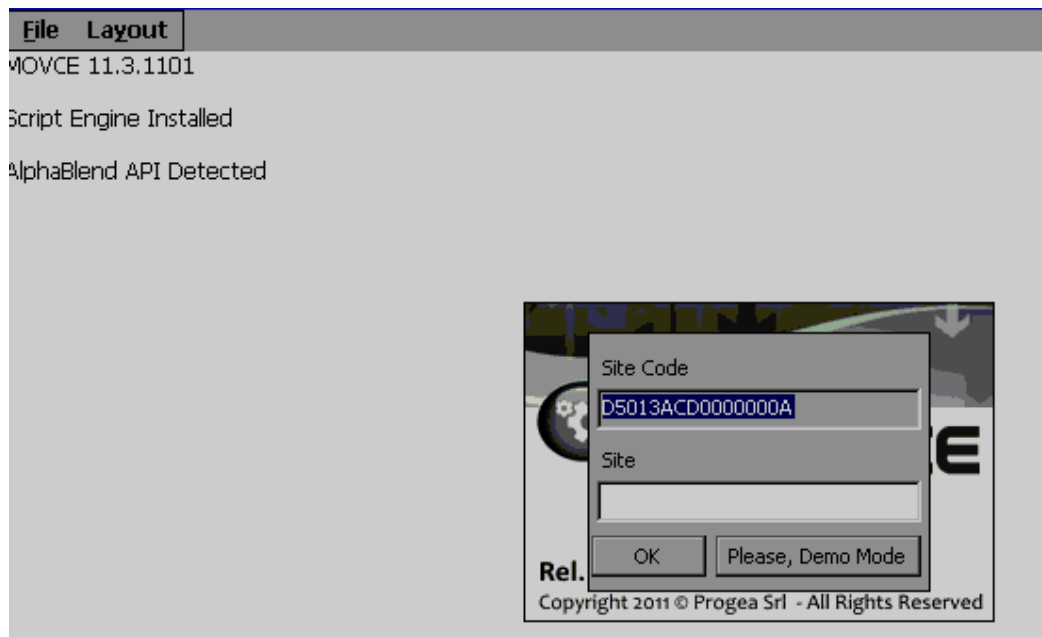
Open ARMV4I and double check the contents



Now, remove the USB disk

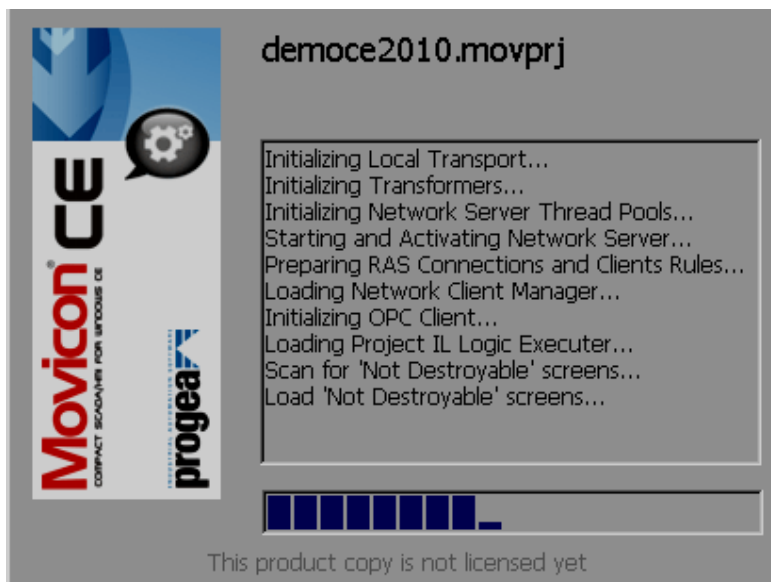
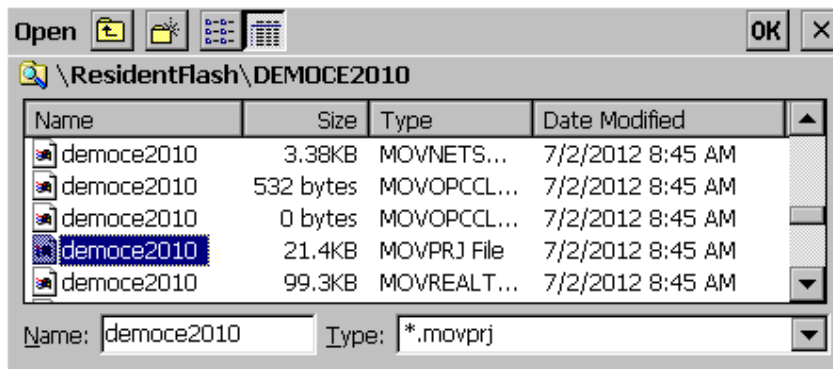
Switch OFF Power supply to HMI

Power ON HMI again



MOVCE.exe start automatically at HMI Power ON as shown above

In case if you don't prepare MovCE.boot file, then, you can open demo project manually as shown below..





How to start project automatically after starting MovCE application

Create MovCE.boot as shown below

```

<?xml version="1.0" encoding="ISO-8859-1" ?>
<boot>
<filename>\ResidentFlash\DEMOCE2010\democe2010.movprj</filename>
<filedate>2012-06-29T16:05:29Z</filedate>
</boot>

```

Copy above file to Resident flash

Note: If Project folder/path is different, then, you have to change MovCE.boot accordingly and copy the file manually to Resident flash

2.1.1.8 Reserved

2.1.1.9 Reserved

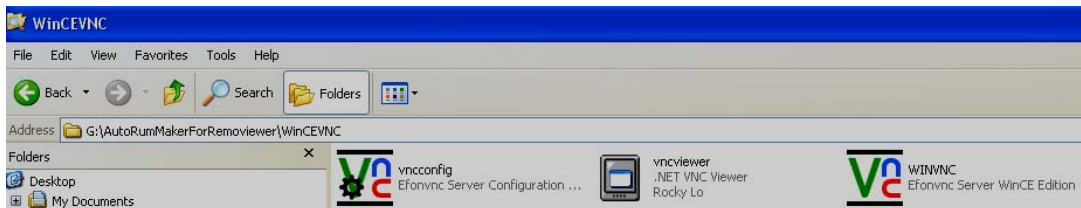
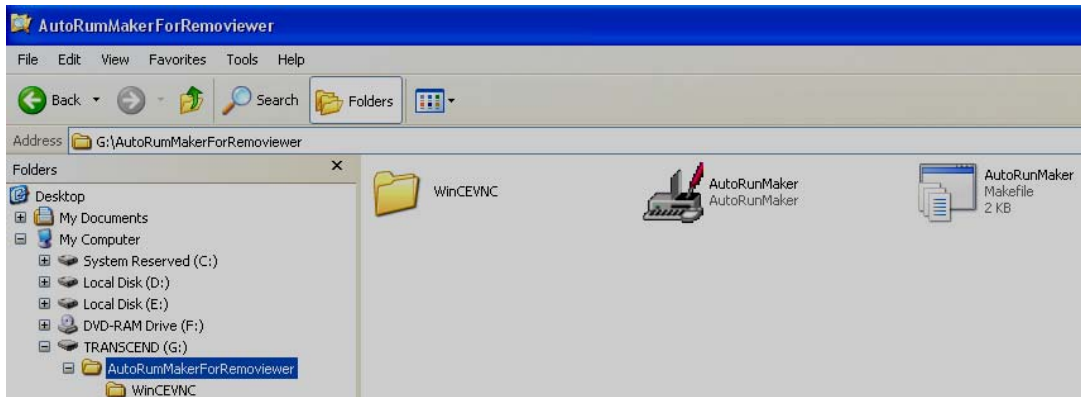
2.1.1.10 How to view HMI screen in PC via VNC

Step-1

Obtain WinCE based VNC tools from the factory

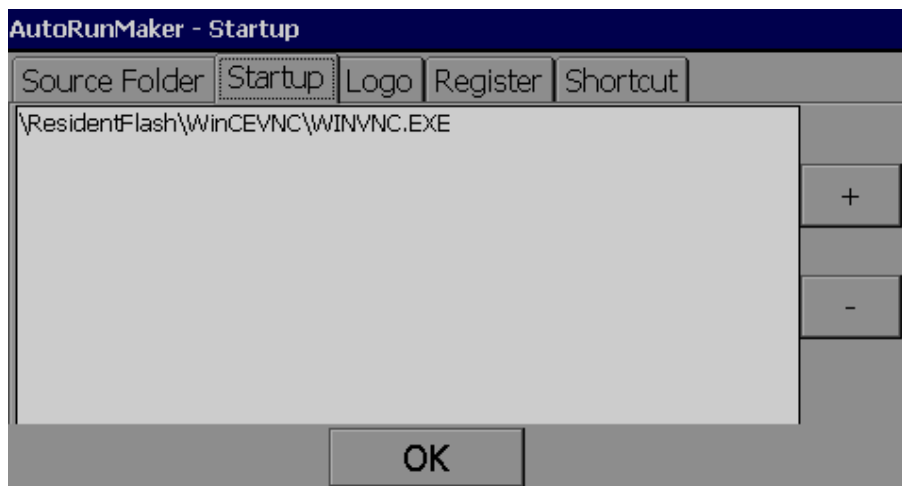
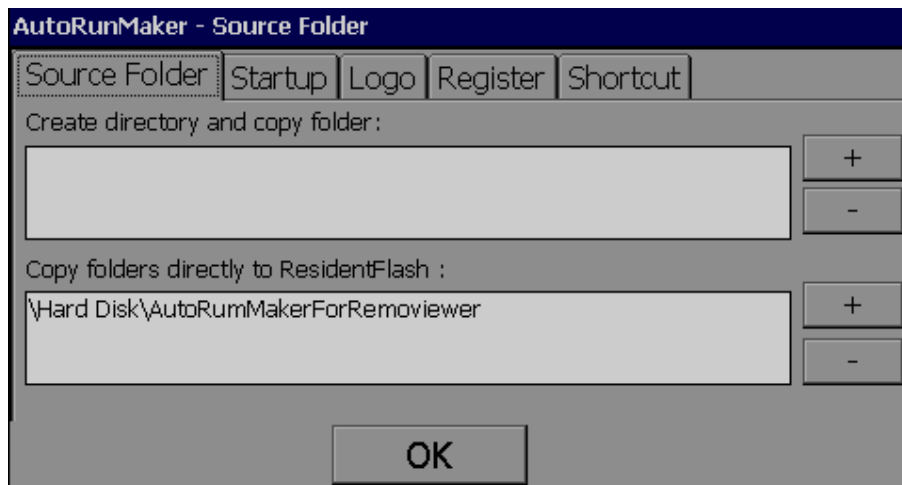
Copy them to USB disk

Contents will look like the following..

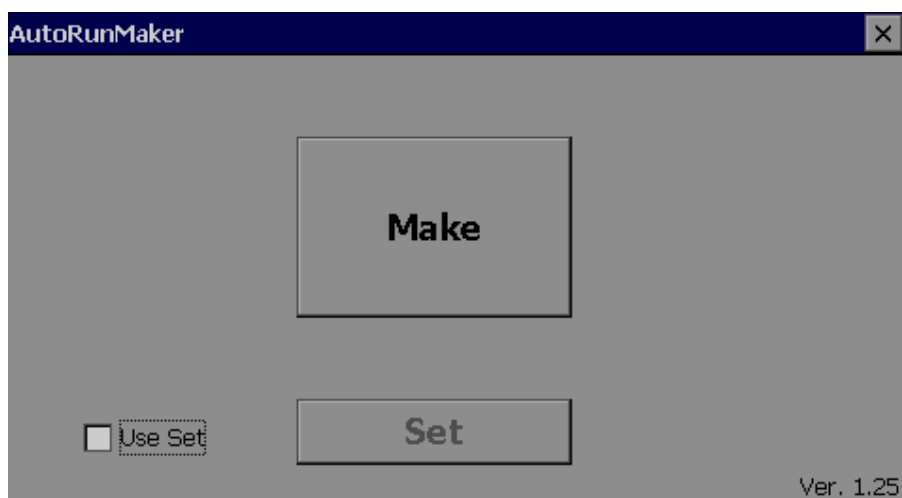


Step-2

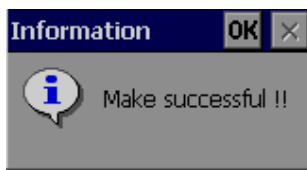
Copy the above files to HMI from the Auto run maker as shown below..



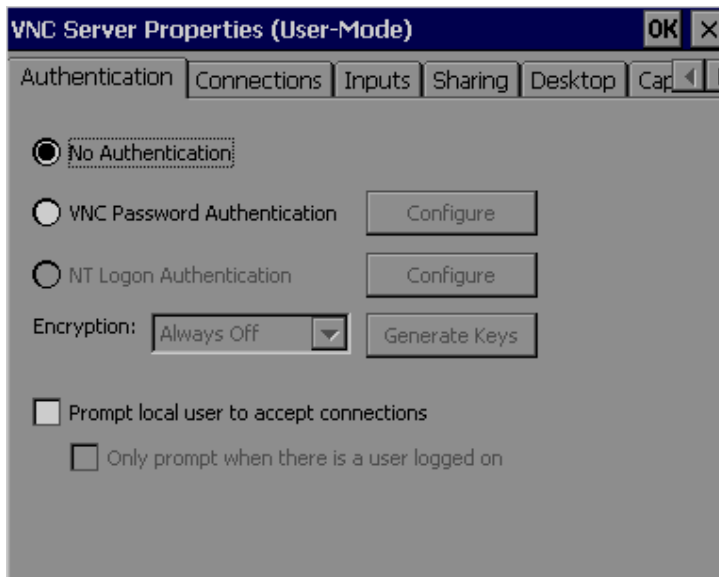
Press "OK"




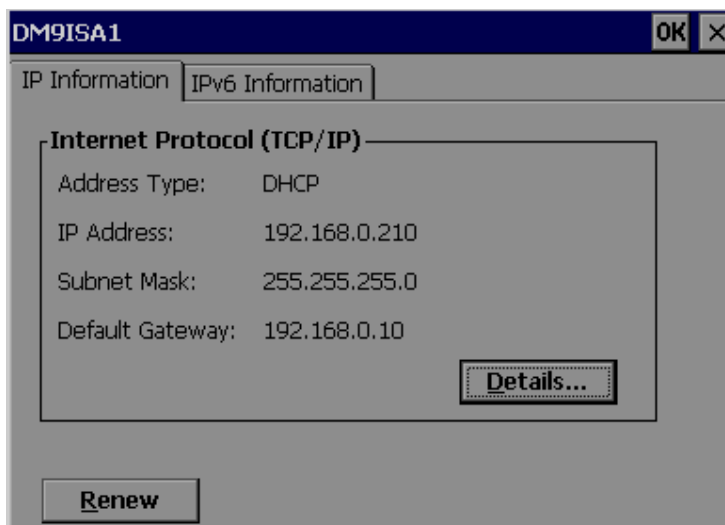
Press "Make"



In the Task bar, open EfonVnc

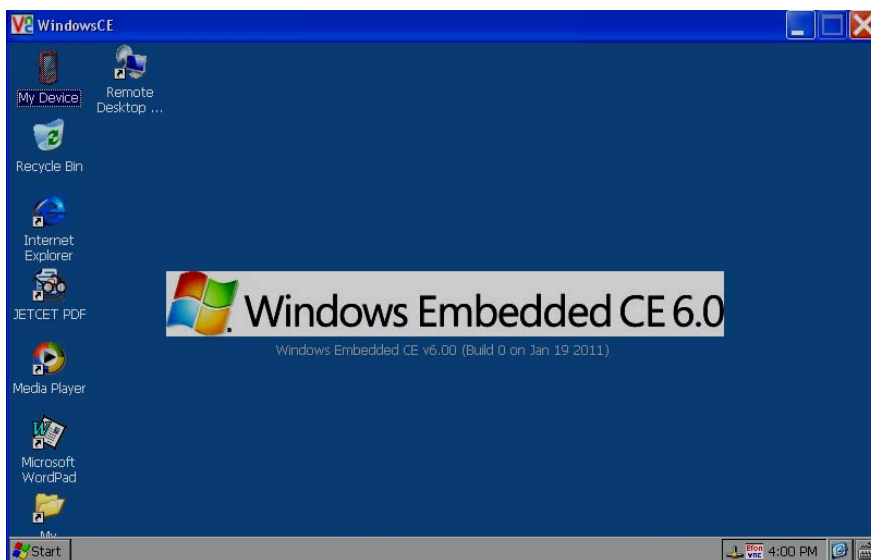
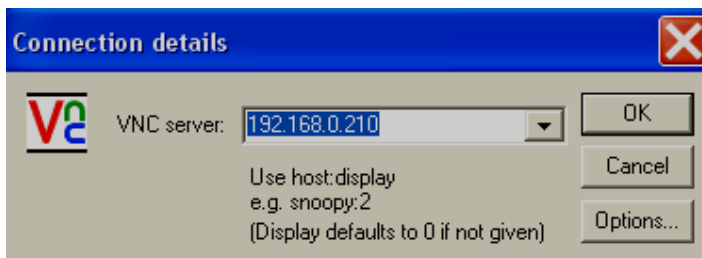
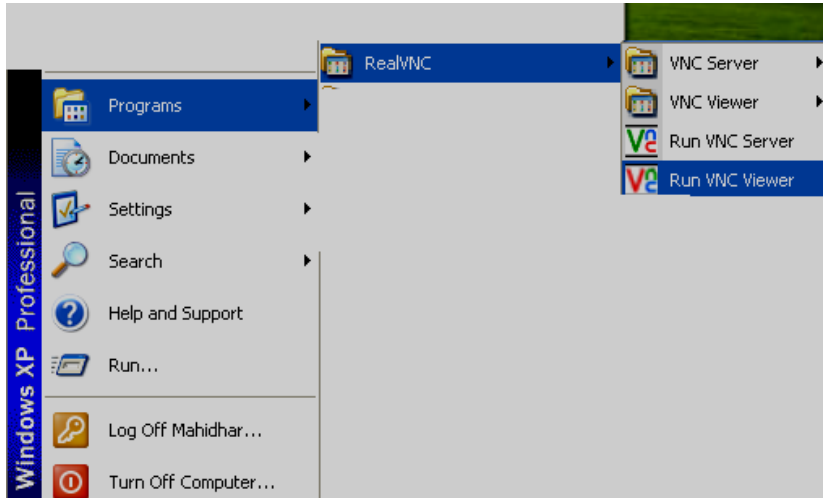


Connect HMI to LAN, open Network icon in the Task bar  and check IP address of the HMI



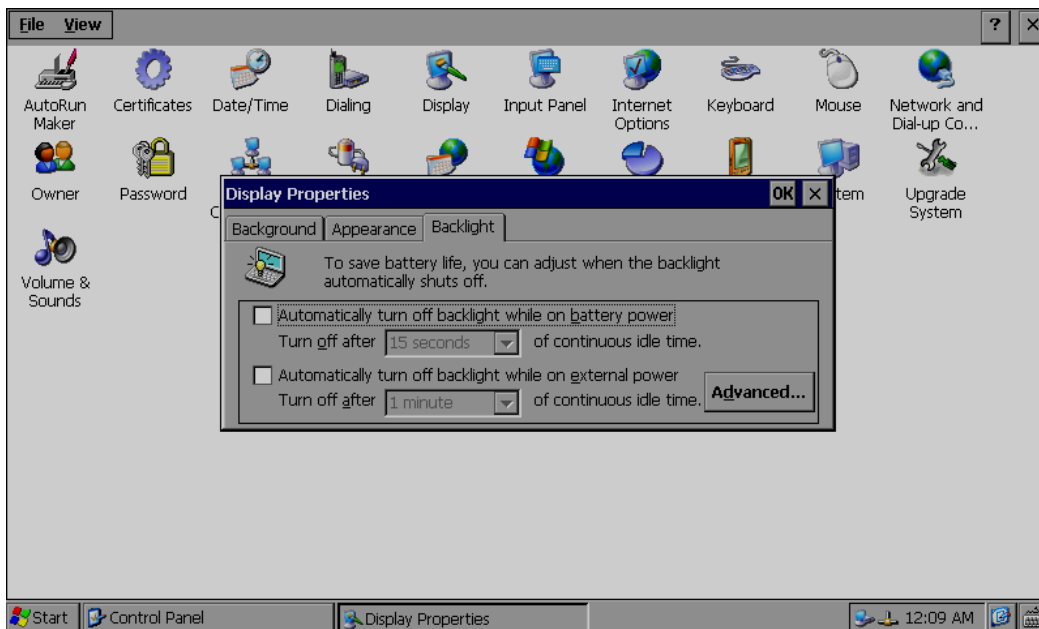
Install VNC software in your PC from the internet.
The file name is CRealVNC337

Then, you can run VNC viewer later as shown below..

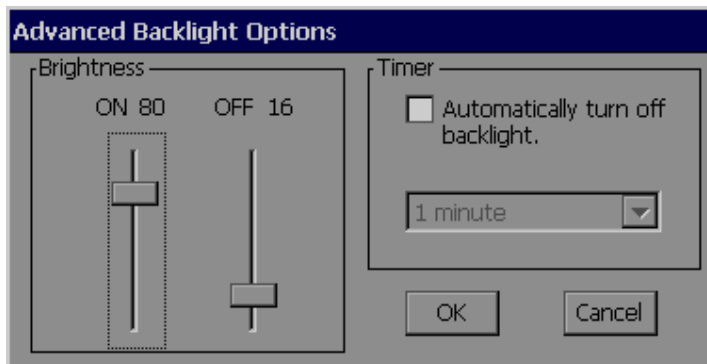


2.1.2 BACKLIGHT ADJUSTMENT

<1> Open Device Control Panel, double-click “Display”, press “Backlight” tab.



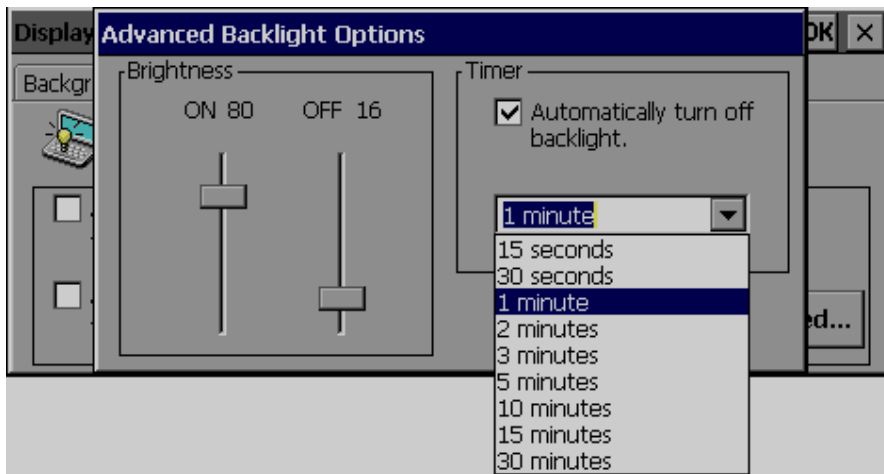
<2> Press “Advanced...” button.
Show dialog “Advanced Backlight Options”.



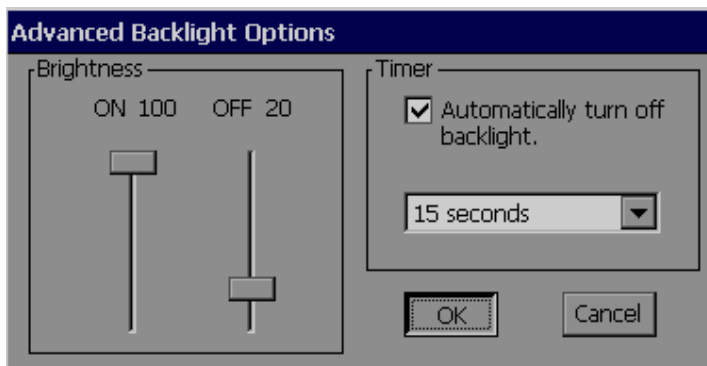
<3> Move slider bar “ON”, set the brightness when backlight on.

<4> Move slider bar “OFF”, set the brightness when backlight off.

<5> Check “Automatically turn off backlight” box to enable backlight off timer. And choose the timeout value.

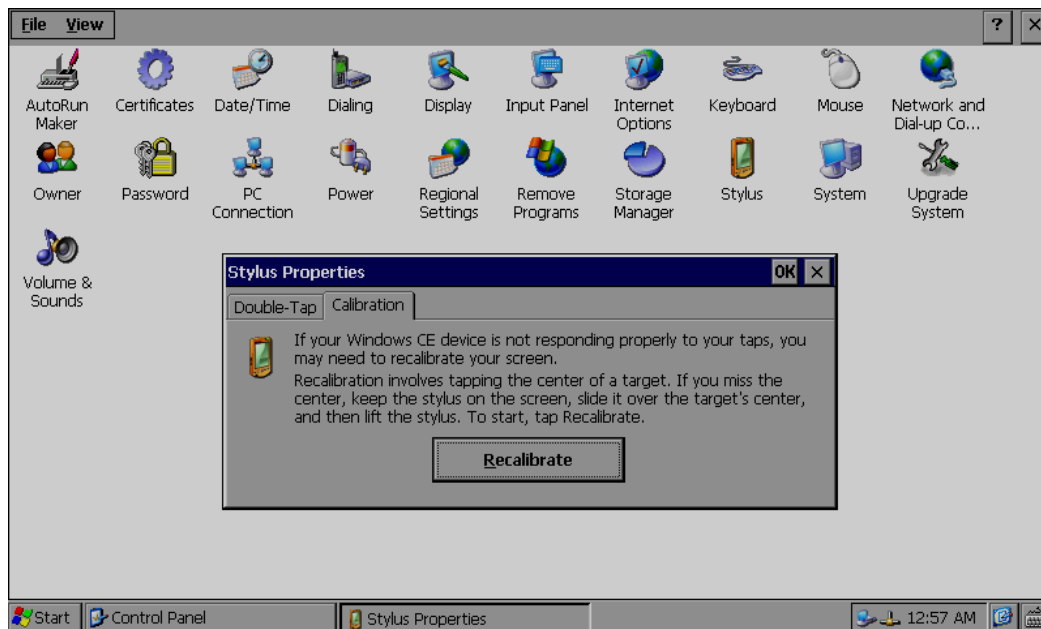


<6> Press “OK” to set all setting. Press “Cancel” to ignore all.



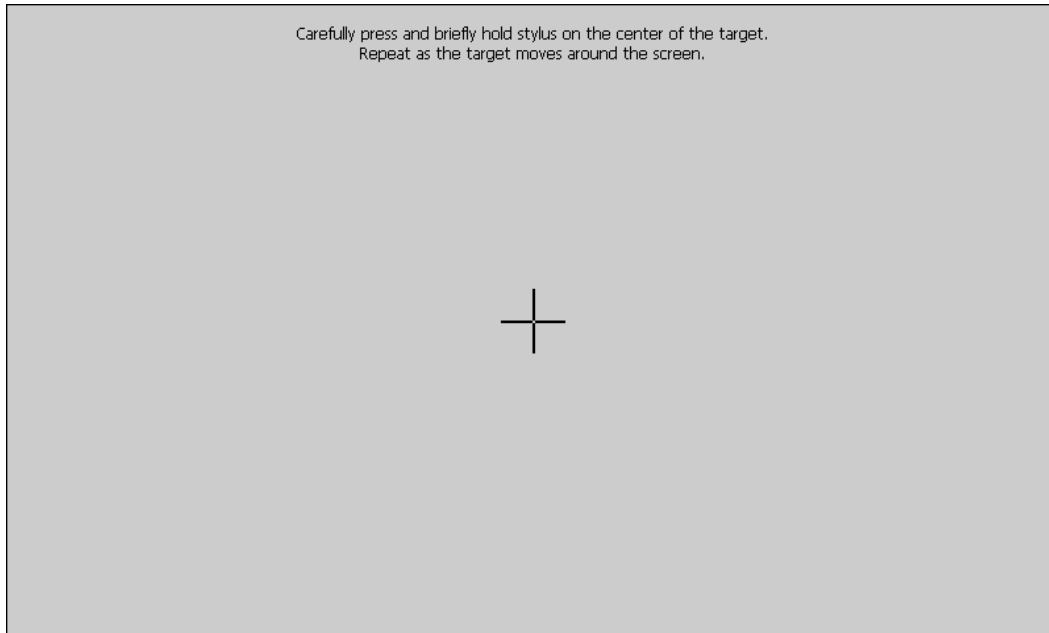
2.1.3 TOUCH PANEL CALIBRATION

<1> Open Device Control Panel, double-click “Stylus”, press “Calibration” tab.



<2> Press "Recalibrate" button.

Carefully press and briefly hold stylus on the center of the target. Repeat as the target moves around the screen.



Carefully press and briefly hold stylus on the center of the target.
Repeat as the target moves around the screen.



Carefully press and briefly hold stylus on the center of the target.
Repeat as the target moves around the screen.



Carefully press and briefly hold stylus on the center of the target.
Repeat as the target moves around the screen.



New calibration settings have been measured.
Touch the screen to register saved data.

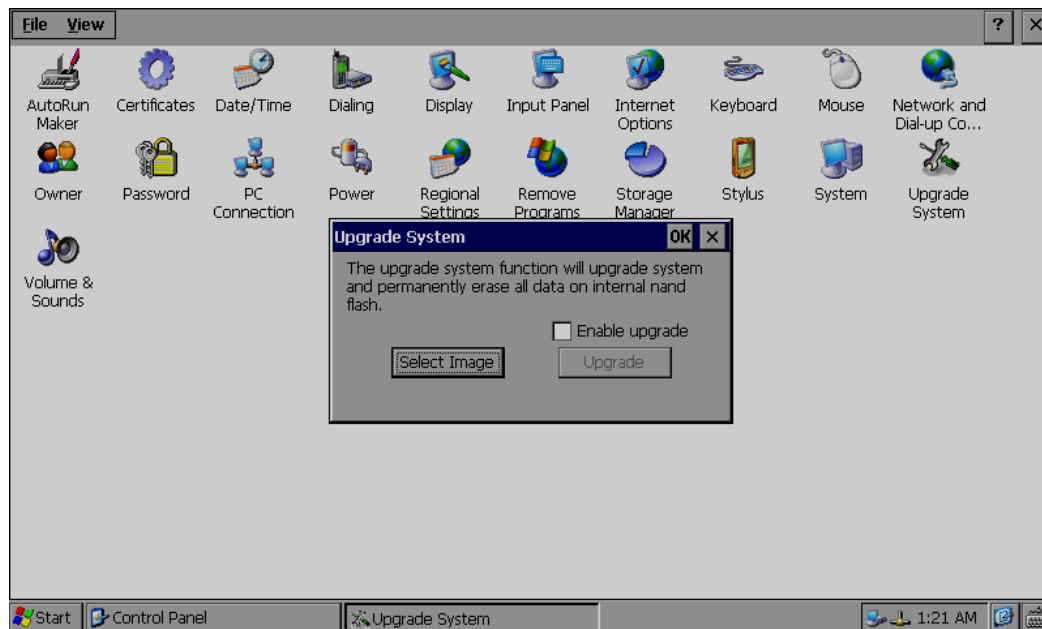
2.1.4 UPGRADE WINDOWS CE IMAGE

It is used to upgrade WinCE image to the latest version.

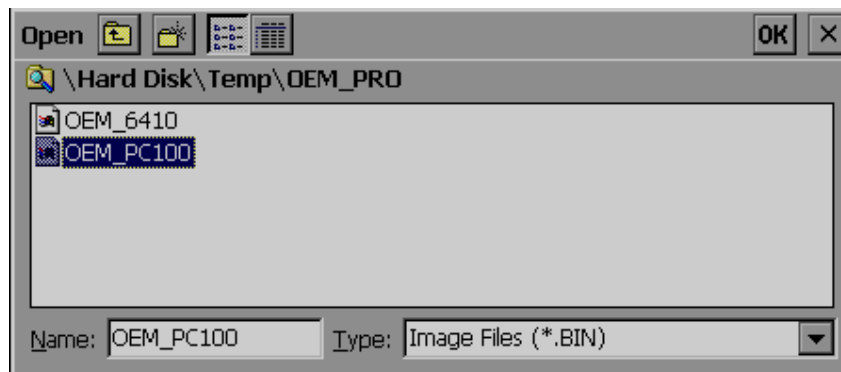
Also, it is used to convert WinCE Core to WinCE professional version and vice versa.

Note: Before converting WinCE Core image to WinCE Professional, you should buy license from Microsoft and attach label on the HMI.

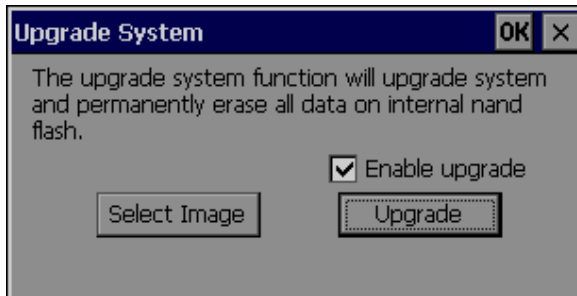
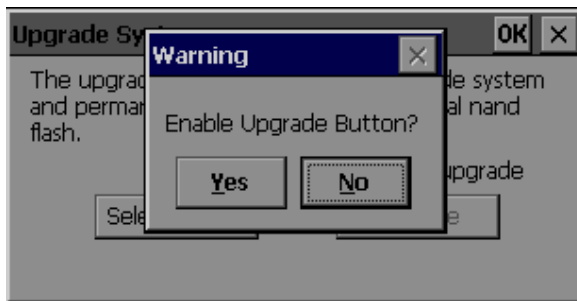
<1> Open Device Control Panel, double-click “Upgrade System”.



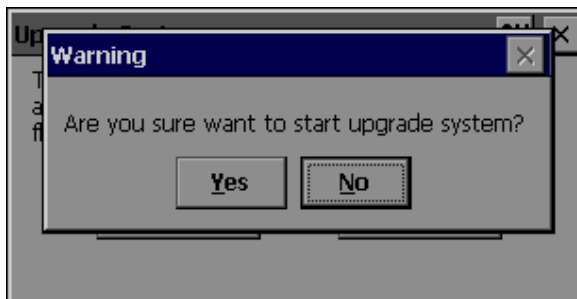
<2> Press “Select Image” button, open update image file. Press “OK”.



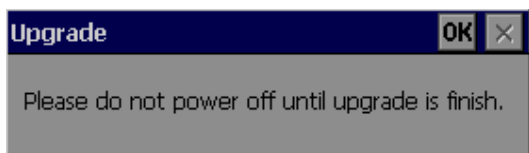
<3> Check "Enable upgrade" box. Press "Yes"



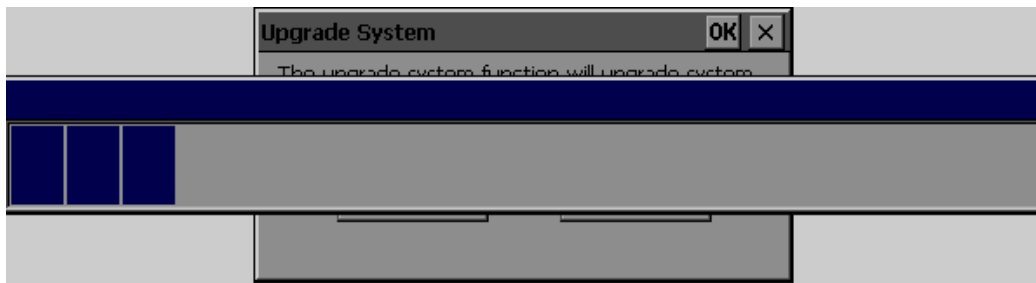
<4> Press "Upgrade" button. Press "Yes".



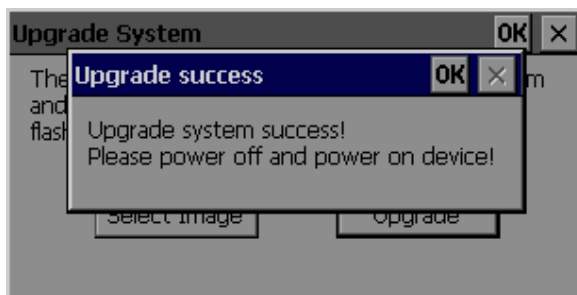
<5> Press "OK".



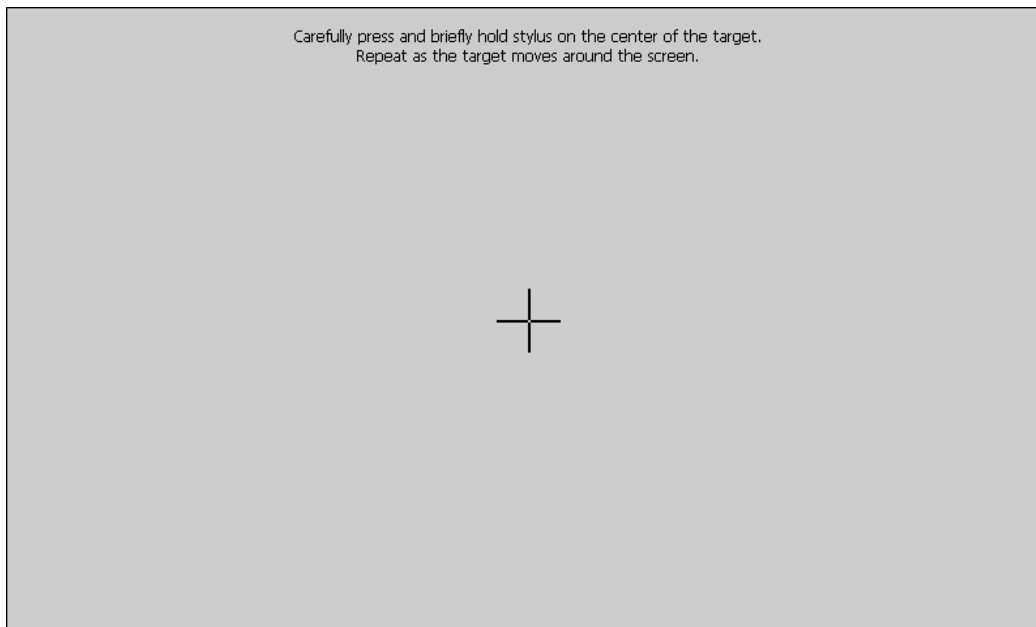
<6> Wait process bar complete.



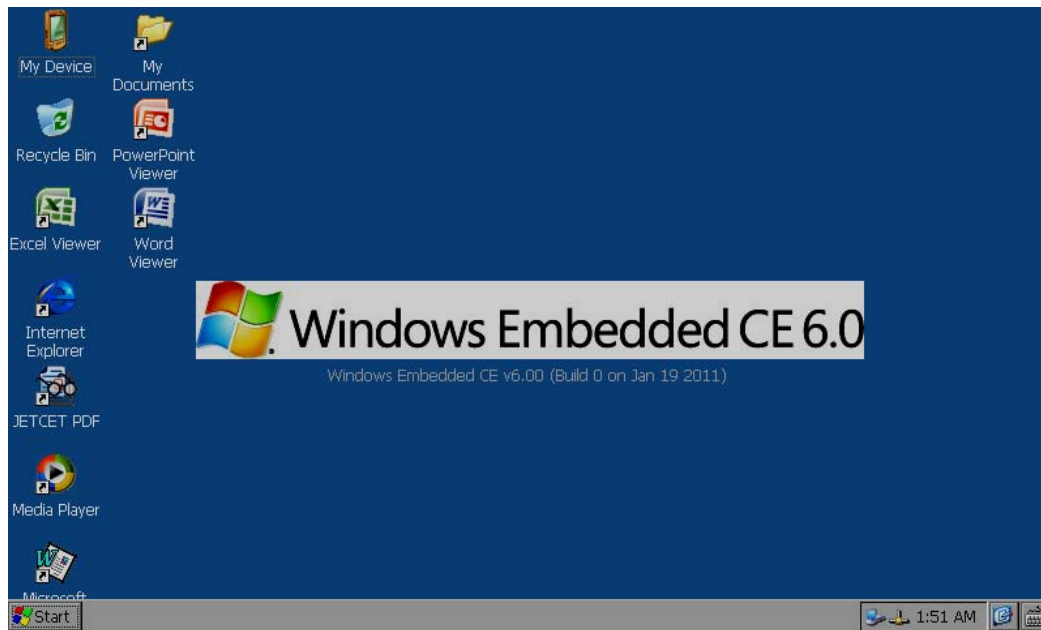
Press "OK".



<7> Power on device again.
The device will auto run touch panel calibration.



<8> Complete touch panel calibration, show desktop.



2.2 WINCE CORE & PROFESSIONAL COMPARISION

- <I> WinCE Core 6410 (HMI 450/730)
- <II> WinCE Core PC100 (HMI 750/1050/1550)
- <III> WinCE Professional 6410Pro (HMI 450/730)
- <IV> WinCE Professional PC100Pro (HMI 750/1050/1550)

Catalog	<I>	<II>	<III>	<IV>
Applications - End User				
ActiveSync	V	V	V	V
File Sync	V	V	V	V
CAB File Installer/Uninstaller	V	V	V	V
WordPad			V	V
File Viewers			V	V
JETCET PDF			V	V
Office Viewers			V	V
Office 2007 Excel Viewer			V	V
Office 2007 PowerPoint Viewer			V	V
Office 2007 Word Viewer			V	V

Applications and Services Development				
.NET Compact Framework 3.5	V	V	V	V
.NET Compact Framework 3.5	V	V	V	V
.NET Compact Framework 3.5 String Resources	V	V	V	V
OS Dependencies for .NET Compact Framework 3.5	V	V	V	V
OS Dependencies for .NET Compact Framework 3.5	V	V	V	V
Active Template Library (ATL)	V	V	V	V
C Libraries and Runtimes	V	V	V	V
C++ Runtime Support for Exception Handling and Runtime Type Information	V	V	V	V
Full C Runtime	V	V	V	V
Standard I/O (STDIO)	V	V	V	V
Standard I/O ASCII (STDIOA)	V	V	V	V
Standard String Functions - ASCII (corestra)	V	V	V	V
Component Services (COM and DCOM)	V	V	V	V
Component Object Model	V	V	V	V
DCOM	V	V	V	V
COM Storage	V	V	V	V
String Safe Utility Functions	V	V	V	V
xML	V	V	V	V
MSxML 3.0	V	V	V	V
xML Core Services and Document Object Model (DOM)	V	V	V	V
xML Query Languages (xQL)	V	V	V	V
xML Stylesheet Language Transformations (xSLT)			V	V
xML Minimal Parser			V	V
Communication Services and Networking				
Networking – General	V	V	V	V
Domain Discovery	V	V	V	V
Extended DNS Querying and Update (DNSAPI)	V	V	V	V
Firewall			V	V
NDIS User-mode I/O Protocol Driver	V	V	V	V
Network Driver Architecture (NDIS)	V	V	V	V
Network Utilities (IpConfig, Ping, Route)	V	V	V	V
TCP/IP	V	V	V	V
IP Helper API	V	V	V	V
TCP/IPv6 Support	V	V	V	V
Windows Networking API/Redirector (SMB/CIFS)	V	V	V	V
Winsock Support	V	V	V	V
Networking - Local Area Network (LAN)	V	V	V	V
Wired Local Area Network (802.3, 802.5)	V	V	V	V

Networking - Wide Area Network (WAN)	V	V	V	V
Dial Up Networking (RAS/PPP)	V	V	V	V
AutoDial	V	V	V	V
Standard Modem Support for Dial Up Networking	V	V	V	V
Telephony API (TAPI 2.0)	V	V	V	V
Unimodem support	V	V	V	V
Servers	V	V	V	V
Core Server Support	V	V	V	V
FTP Server	V	V	V	V
Simple Network Time Protocol (SNTP)	V	V	V	V
SNTP Client with DST	V	V	V	V
Web Proxy			V	V
Web Server (HTTPD)			V	V
Active Server Pages (ASP) Support			V	V
JScript 5.6			V	V
VBScript 5.6			V	V
Web Server Administration ISAPI			V	V
WebDAV Support			V	V
Core OS Services				
Battery Driver	V	V	V	V
Debugging Tools	V	V	V	V
Toolhelp API	V	V	V	V
Device Manager	V	V	V	V
Display Support	V	V	V	V
Kernel Functionality	V	V	V	V
Fiber API			V	V
FormatMessage API	V	V	V	V
FormatMessage API - System Error Messages			V	V
Memory Mapped Files	V	V	V	V
Message Queue - Point-to-Point	V	V	V	V
Target Control Support (Shell.exe)	V	V	V	V
Notification (Choose 1)	V	V	V	V
UI based Notification	V	V	V	V
Power Management (Choose 1)	V	V	V	V
Power Management (Full)	V	V	V	V
Serial Port Support	V	V	V	V
UI Proxy for Kernel-Mode Drivers	V	V	V	V
USB Host Support	V	V	V	V
USB Function Driver	V	V	V	V
USB Host Support	V	V	V	V
USB Human Input Device (HID) Class Driver	V	V	V	V

USB HID Keyboard and Mouse	V	V	V	V
USB HID Keyboard Only	V	V	V	V
USB HID Mouse Only	V	V	V	V
USB Printer Class Driver	V	V	V	V
USB Storage Class Driver	V	V	V	V
Windows Embedded CE Driver Development Kit Support Library	V	V	V	V
Device Management				
File Systems and Data Store				
Compression	V	V	V	V
Database Support	V	V	V	V
CEDB Database Engine	V	V	V	V
File and Database Replication	V	V	V	V
Bit-based	V	V	V	V
File Cache Manager	V	V	V	V
File System - Internal (Choose 1)	V	V	V	V
RAM and ROM File System	V	V	V	V
Registry Storage (Choose 1)	V	V	V	V
Hive-based Registry	V	V	V	V
Storage Manager	V	V	V	V
Binary Rom Image File System	V	V	V	V
exFAT File System	V	V	V	V
FAT File System	V	V	V	V
Partition Driver	V	V	V	V
Release Directory File System	V	V	V	V
Storage Manager Control Panel Applet	V	V	V	V
Transaction-Safe FAT File System (TFAT)	V	V	V	V
System Password	V	V	V	V
Fonts				
Comic Sans MS	V	V	V	V
Comic Sans MS	V	V	V	V
Symbol	V	V	V	V
Times New Roman	V	V	V	V
Times New Roman (Subset 1_30)	V	V	V	V
Wingding			V	V
Graphics and Multimedia Technologies				
Audio		V	V	V
Waveform Audio		V	V	V
Graphics	V	V	V	V
Alphablend API (GDI version)			V	V
Direct3D Mobile			V	V

DirectDraw			V	V
Gradient Fill Support	V	V	V	V
Imaging	V	V	V	V
Still Image Codec Support (Encode and Decode)	V	V	V	V
Still Image Decoders	V	V	V	V
BMP Decoder	V	V	V	V
GIF Decoder	V	V	V	V
JPG Decoder	V	V	V	V
PNG Decoder	V	V	V	V
Still Image Encoders	V	V	V	V
BMP Decoder	V	V	V	V
GIF Decoder	V	V	V	V
JPG Decoder	V	V	V	V
PNG Decoder	V	V	V	V
Media				
Audio Codecs and Renderers			V	V
MP3 Codec			V	V
Wave/AIFF/au/snd File Parser			V	V
WMA Codec			V	V
WMA Voice Codec			V	V
DirectShow			V	V
ACM Wrapper Filter			V	V
DirectShow Core			V	V
DirectShow Display			V	V
DirectShow Video Capture			V	V
DMO Wrapper Filter			V	V
Media Formats			V	V
MPEG-1 Parser/Splitter			V	V
Video Codecs and Renderers			V	V
DirectShow Video Renderer			V	V
WMV/MPEG-4 Video Codec			V	V
Windows Media Player			V	V
Windows Media Player			V	V
Windows Media Player OCX			V	V
Windows Media Technologies			V	V
ASx v1 and M3U File Support			V	V
ASx v2 File Support			V	V
ASx v3 File Support			V	V
NSC File Support			V	V
Windows Media Multicast and Multi-Bit Rate			V	V
Windows Media Streaming from Local Storage			V	V

Windows Media Streaming over HTTP			V	V
Windows Media Streaming over MMS			V	V
WMA and MP3 Local Playback			V	V
WMA and MP3 Streaming			V	V
International				
Input Method Manager (IMM)	V	V	V	V
Locale Services (Choose 1)	V	V	V	V
National Language Support (NLS)	V	V	V	V
Locale Specific Support	V	V	V	V
Arabic	V	V	V	V
Fonts	V	V	V	V
Arial (subset 1_08)	V	V	V	V
Arial Bold (subset 1_08)	V	V	V	V
Courier New (subset 1_08)	V	V	V	V
Tahoma (subset 1_08)	V	V	V	V
Tahoma Bold (subset 1_08)	V	V	V	V
Unicode Script Processor for Complex Scripts	V	V	V	V
Chinese (Simplified)	V	V	V	V
Monotype Imaging AC3 Font Compression	V	V	V	V
Chinese (Traditional)	V	V	V	V
Monotype Imaging AC3 Font Compression	V	V	V	V
Hebrew	V	V	V	V
Fonts	V	V	V	V
Arial (subset 1_08)	V	V	V	V
Arial Bold (subset 1_08)	V	V	V	V
Courier New (subset 1_08)	V	V	V	V
Tahoma (subset 1_08)	V	V	V	V
Unicode Script Processor for Complex Scripts	V	V	V	V
Indic	V	V	V	V
Unicode Script Processor for Complex Scripts	V	V	V	V
Japanese	V	V	V	V
Monotype Imaging AC3 Font Compression	V	V	V	V
Korean	V	V	V	V
Monotype Imaging AC3 Font Compression	V	V	V	V
Thai	V	V	V	V
Fonts	V	V	V	V
Tahoma (subset 1_08)	V	V	V	V
Unicode Script Processor for Complex Scripts	V	V	V	V
Internet Client Services				
Browser Application			V	V

Internet Explorer 6.0 Windows Embedded CE - Standard Components			V	V
Internet Explorer 6.0 Sample Browser			V	V
Internet Explorer 6.0 for Windows Embedded CE Components	V	V	V	V
Internet Explorer Browser Control Host			V	V
Internet Explorer HTML/DHTML API			V	V
Filter and Translation			V	V
Internet Explorer HTML Application			V	V
Internet Explorer Plug-in Image Decoder API			V	V
Internet Explorer PNG Image Decoder			V	V
Internet Explorer Theme Library			V	V
MSHTML Data Binding			V	V
Internet Explorer Multiple-Language Base API			V	V
Internet Explorer Multiple-Language Full API			V	V
Optional Charset/Encoding in registry			V	V
URL Moniker Services			V	V
Windows Internet Services	V	V	V	V
Passport SSI 1.4 Authentication			V	V
Platform for Privacy Preferences (P3P)			V	V
xML Data Islands			V	V
xML MIME Viewer			V	V
Internet Options Control Panel			V	V
Scripting			V	V
JScript 5.6			V	V
Script Authoring (Jscript)			V	V
Script Encode (Jscript)			V	V
VBScript 5.6			V	V
MsgBox and InputBox support			V	V
Script Authoring (VBScript)			V	V
Script Encode (VBScript)			V	V
Security				
Authentication Services (SSPI)	V	V	V	V
Kerberos	V	V	V	V
NTLM	V	V	V	V
Schannel (SSL/TLS)	V	V	V	V
Credential Manager	V	V	V	V
Cryptography Services (CryptoAPI 1.0) - High Encryption Provider	V	V	V	V
Certificates (CryptoAPI 2.0)	V	V	V	V
Shell and User Interface				
Graphics, Windowing and Events	V	V	V	V

Minimal GDI Configuration	V	V	V	V
Minimal GWES Configuration	V	V	V	V
Minimal Input Configuration	V	V	V	V
Minimal Window Manager Configuration	V	V	V	V
Shell	V	V	V	V
Command Shell	V	V	V	V
Command Processor	V	V	V	V
Console Window	V	V	V	V
Graphical Shell (Choose 1)	V	V	V	V
Standard Shell	V	V	V	V
User Interface	V	V	V	V
Common Controls	V	V	V	V
Common Control	V	V	V	V
Common Dialog Support	V	V	V	V
Control Panel Applets	V	V	V	V
Mouse	V	V	V	V
Network User Interface	V	V	V	V
Software Input Panel	V	V	V	V
Software-based Input Panel (SIP) (Choose 1 or more)	V	V	V	V
SIP for Large Screens	V	V	V	V
SIP for Small Screens	V	V	V	V
Software-based Input Panel Driver	V	V	V	V
Touch Screen (Stylus)	V	V	V	V
Voice over IP Phone Services				
Windows Embedded CE Error Reporting				

2.3 APPENDIX

Sample6410.pbxml component list
 SamplePC100.pbxml component list
 Sample6410Pro.pbxml component list
 SamplePC100Pro.pbxml component list

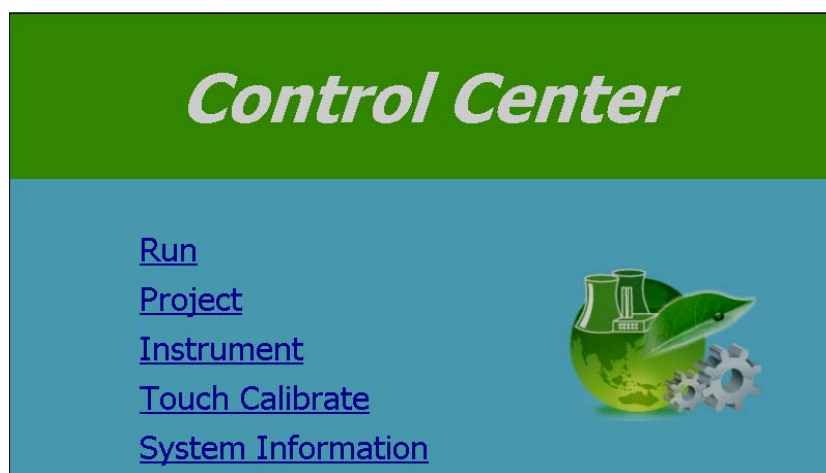
2.4 WINCE IMAGE CONVERSION

2.4.1 Control Center V1.1x to WinCE Professional

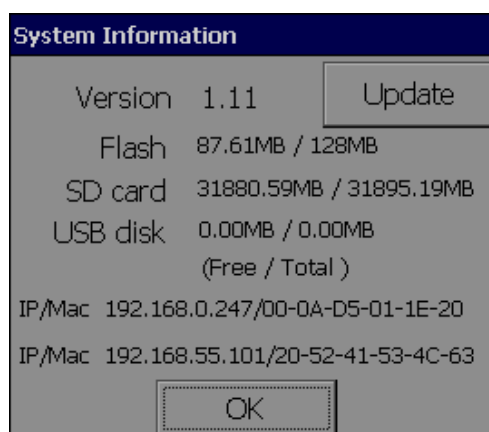


Please remember that, if you replace image from WinCE Core to WinCE Professional, you must buy license from Microsoft and then replace label on back side of the HMI enclosure

1. Check firmware version in HMI first



Press "System Information"



Make sure, it is V1.1x

2. Contact factory for latest WinCE Pro. Image files and update files.

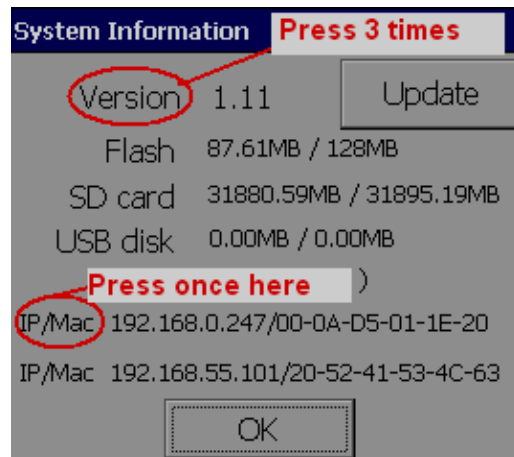
Update files: UpgradeSystem.exe & IOAPI.dll

WinCE Professional version image files:

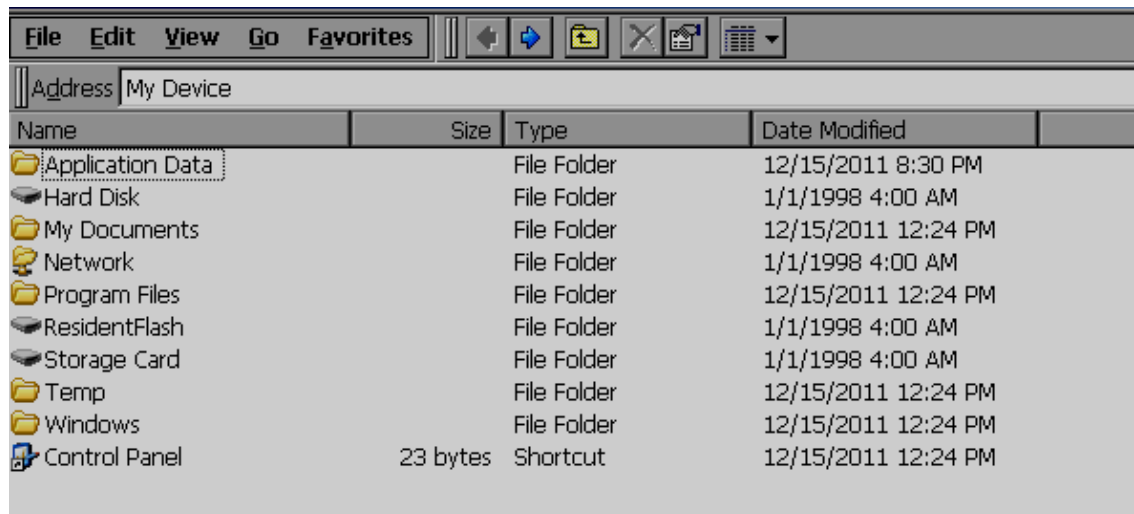
OEM_6410 for HMI 450/730

OEM_PC100 for HMI 750/1050/1550

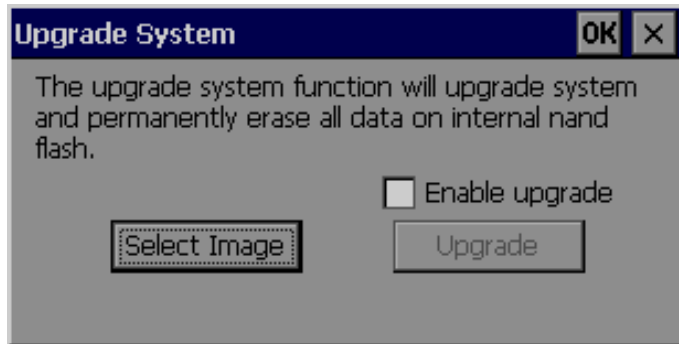
3. Take one empty USB flash disk. We suggest to format USB disk and then copy all the required update files and WinCE image files into USB
4. Now insert USB disk into HMI
5. In System information screen, press three times on top of string "Version" and one time on top of string "IP/Mac" as shown below. Then, it will go to Windows CE desktop



6. Following screen appears in HMI



7. Open "Hard disk" and then run "UpdateSystem.exe"



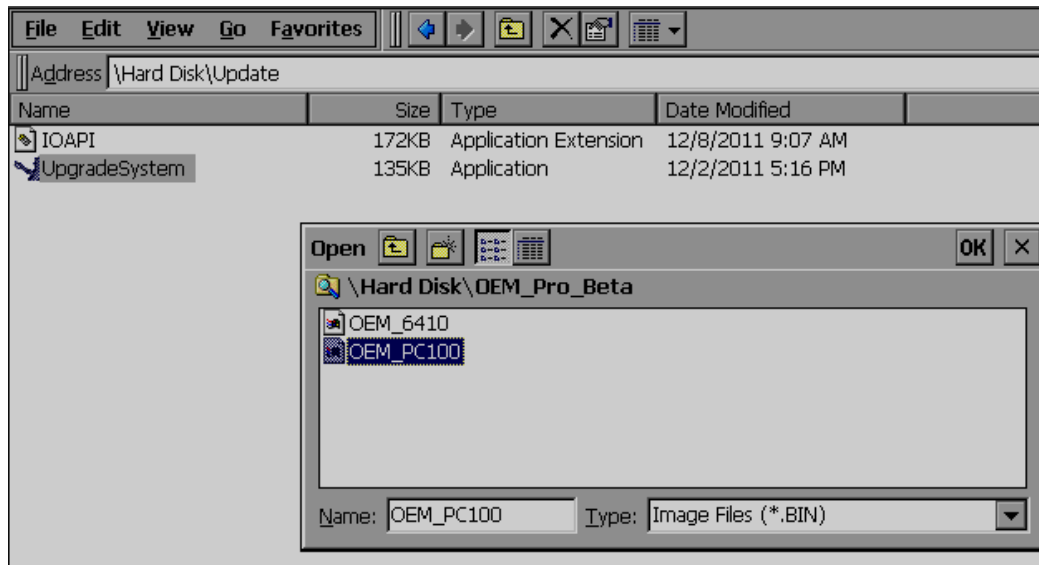
Press "Select Image"

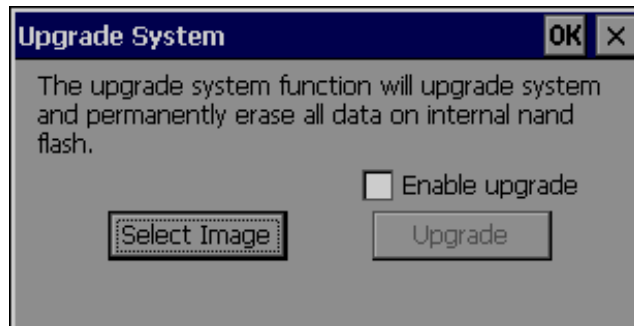


Make sure you select correct images as follows

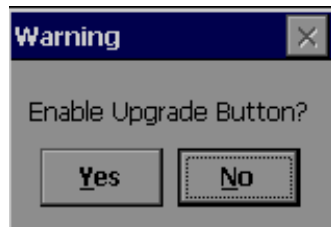
OEM_6410 for HMI 450/730

OEM_PC100 for HMI 750/1050/1550

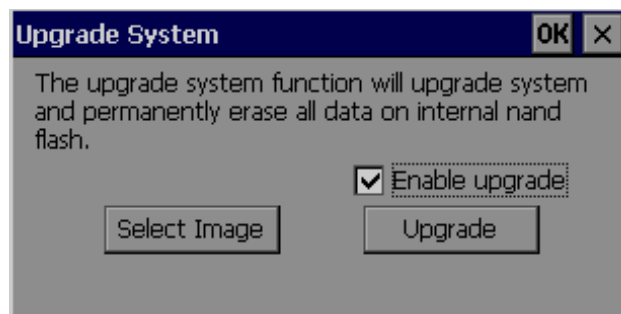




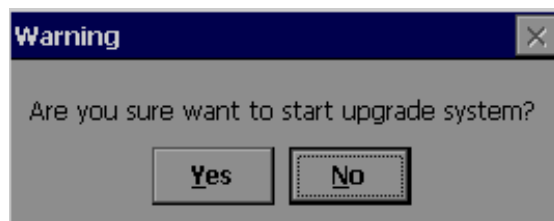
Select "Enable upgrade"



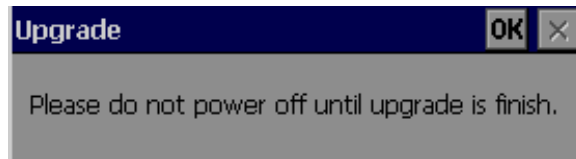
Press "Yes"



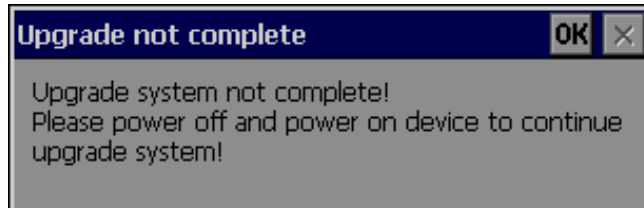
Now, Upgrade button is enabled. Press "Upgrade" button



Press "Yes"



Press "OK"



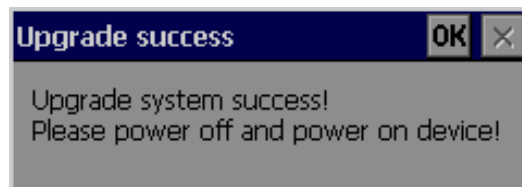
After Power ON, it shows a message "Initializing wait a moment". Now, at this time, it will change partition size from 32 MB to latest size. But, all the image update process is not yet completed.

If prompted, complete the calibration for the touch screen. Then, it goes again to "Control Center" screen. Now repeat the process from Step-5 again.



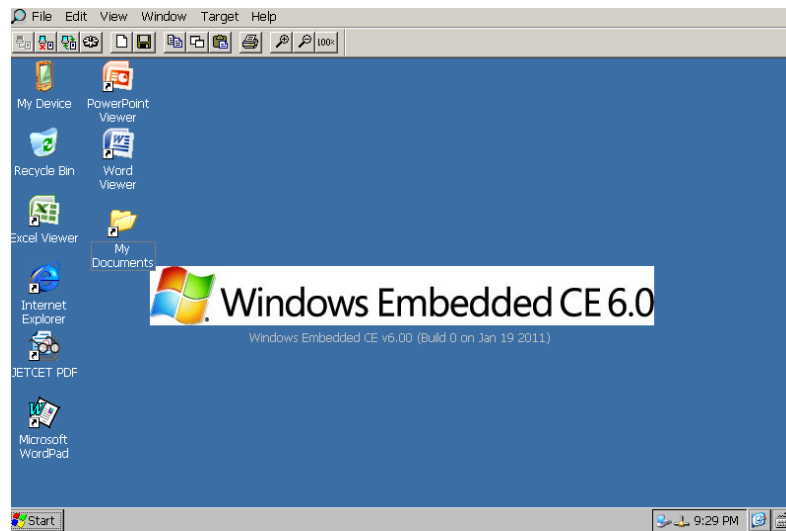
Do not Power OFF HMI during this process till loading bar is completed and display next message about update status. If Power supply is disconnected during this process, you will be not able to recover HMI and you should return HMI back to factory

If image update process is successful, the following message will be displayed in HMI



Press "OK". Now, Switch OFF Power supply and Power ON again

If prompted, complete the calibration for the touch screen. Then, it goes again to "Windows CE desk top



3. BOARD SUPPORT PACKAGE (BSP)

It is required only for companies who are having engineers familiar with WinCE operating system & wish to prepare their own WinCE image. In this case, we expect to receive WinCE image file back to us. We will load the image into HMI before shipment. BSP is not required for every one. Since some of the tools already available to select a path to start an application at Power On, boot UP logo etc., you can try to use our standard WinCE core or Professional image itself first and if you need many other features, then, you may need to customize WinCE image by yourself using BSP

3.1 Software requirements

Microsoft Visual Studio 2005 Service Pack 1
Windows Embedded CE 6.0 R3

3.2 Install BSP SMDK6410

This is for HMI 450 and HMI 730 only

Quick Procedure

1. Decompress the BSP compression file.(SMDK6410_1V2B11.rar)
2. Copy folder "SMDK6410" to "C:\WINCE600\PLATFORM"
3. Under "C:\WINCE600\OSDesigns" create a project folder.
For example: "C:\WINCE600\OSDesigns\MyDevice".
4. Copy "Sample6410.pbxml" to folder "C:\WINCE600\OSDesigns\MyDevice".
5. Double click "Sample6410.pbxml".
6. Change build mode : "SMDK6410 ARMV4I Release"
7. Select "Sample6410", press mouse right button, select "properties".

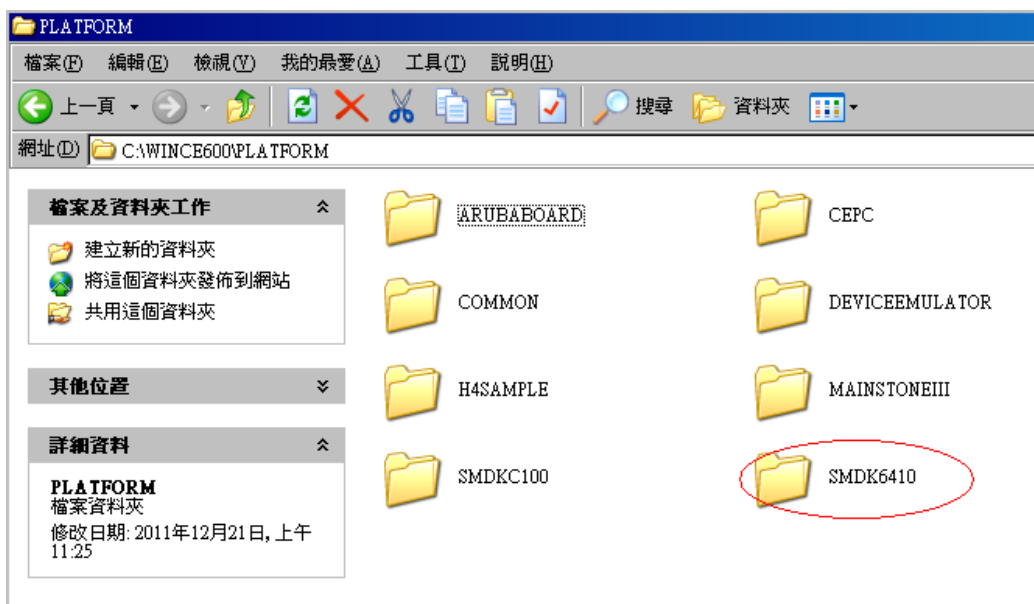
Select "Build Options", uncheck "Enable kernel debugger" ,
uncheck "Enable KITL" , check "Run-time image can be larger than 32MB"
8. Select "Sample6410", press mouse right button, select "Build Sample6410"

Detailed Procedure

<1> Decompress the BSP compression file.(SMDK6410_1V2B11.rar)

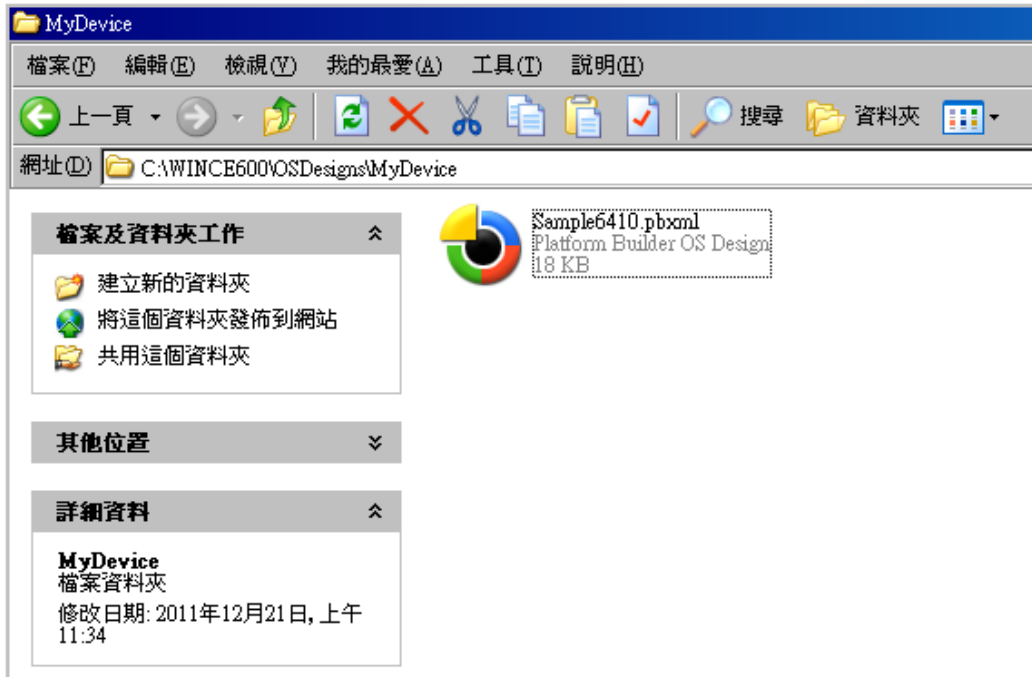


<2> Copy folder “SMDK6410” to “C:\WINCE600\PLATFORM”.



<3> Under “C:\WINCE600\OSDesigns” create a project folder.
For example: “C:\WINCE600\OSDesigns\MyDevice”.

<4> Copy “Sample6410.pbxml” to folder
“C:\WINCE600\OSDesigns\MyDevice”.

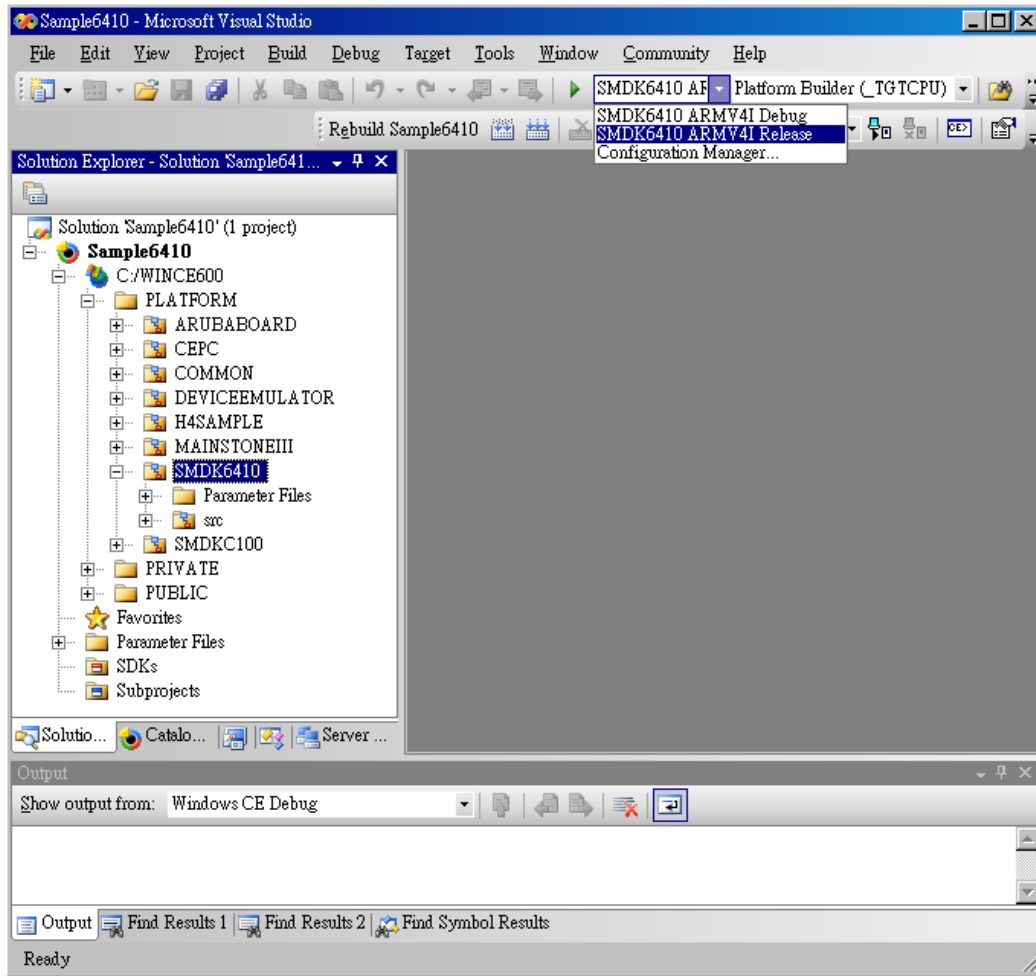


<5> Double click "Sample6410.pbxml".

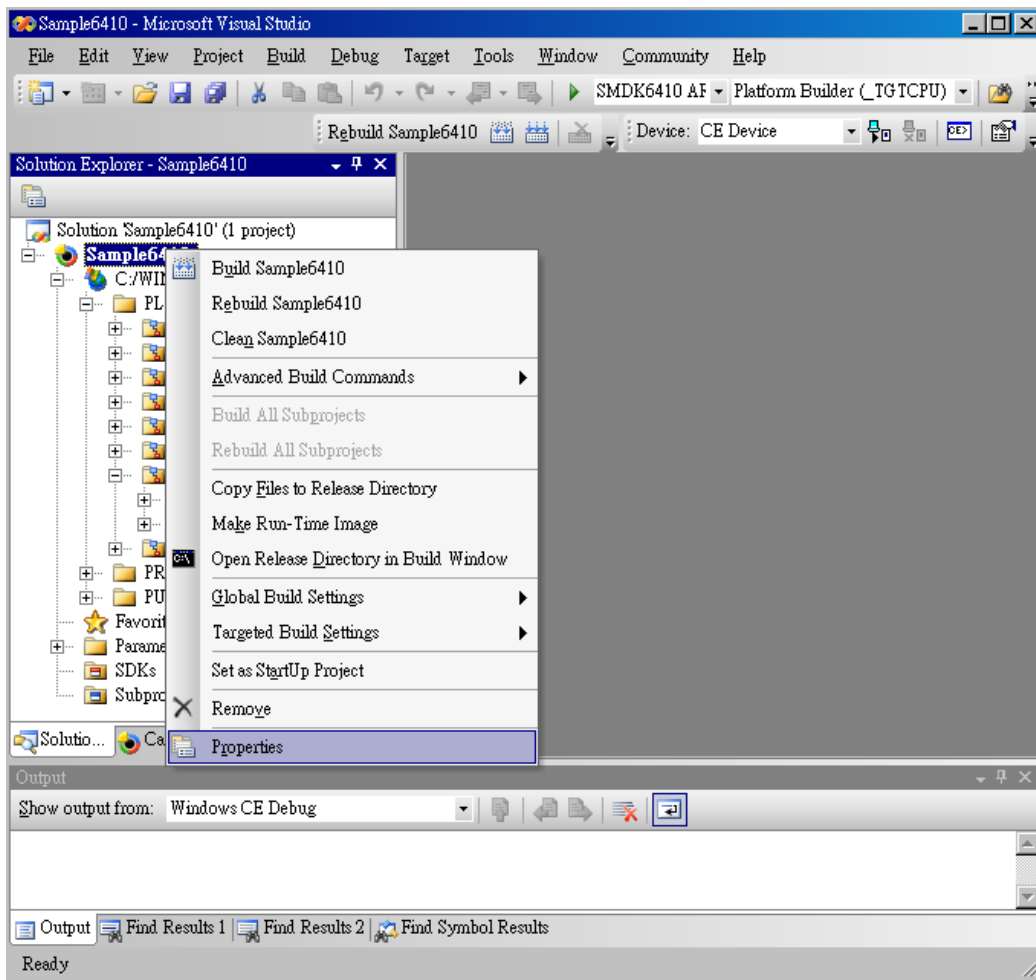
Then Microsoft Visual Studio 2005 start to run.

<6> Change build mode : "SMDK6410 ARMV4I Release"

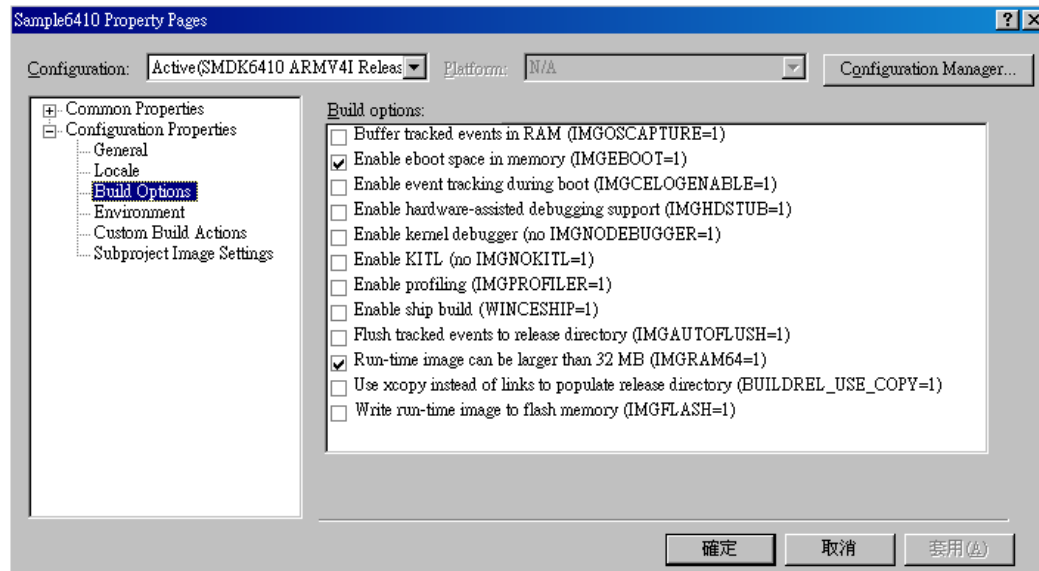
Expend "Sample6410-C:/WINCE600-PLATFORM-SMDK6410"



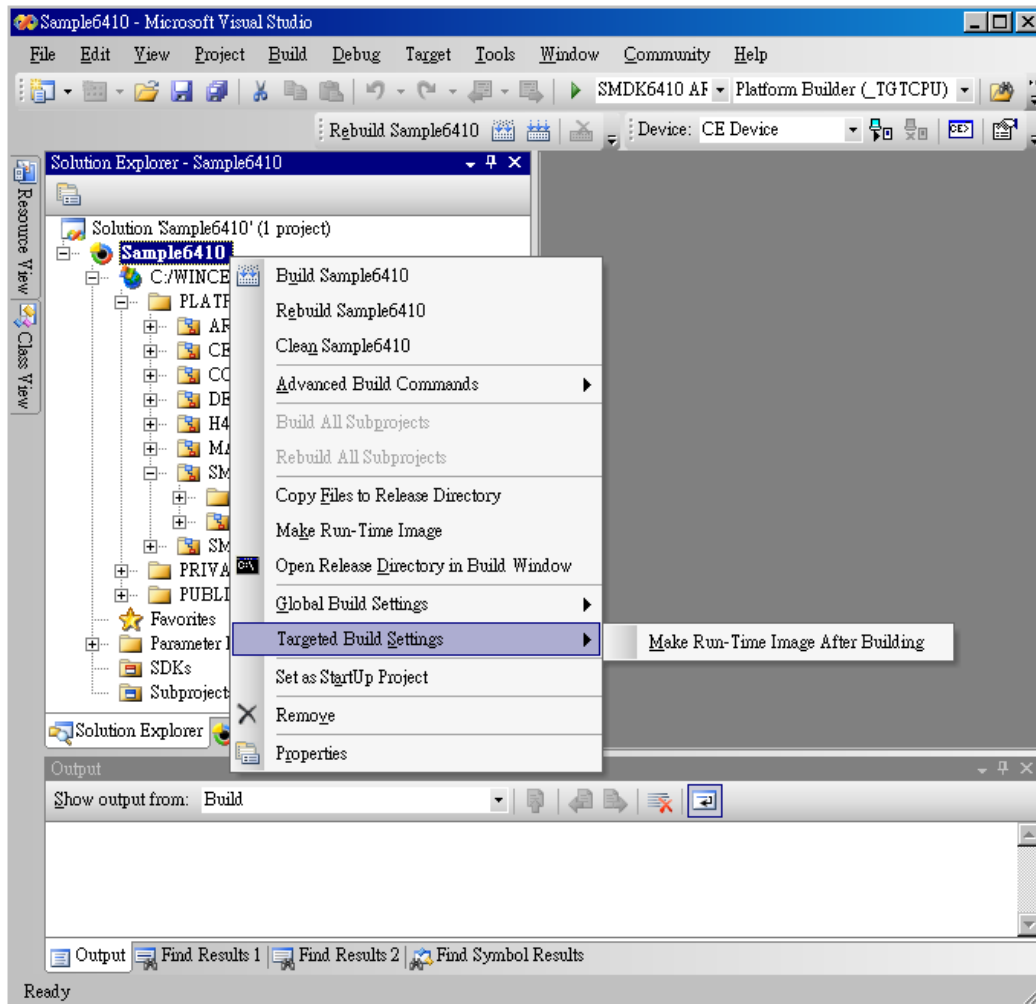
<7> Select “Sample6410”, press mouse right button, select “properties”.



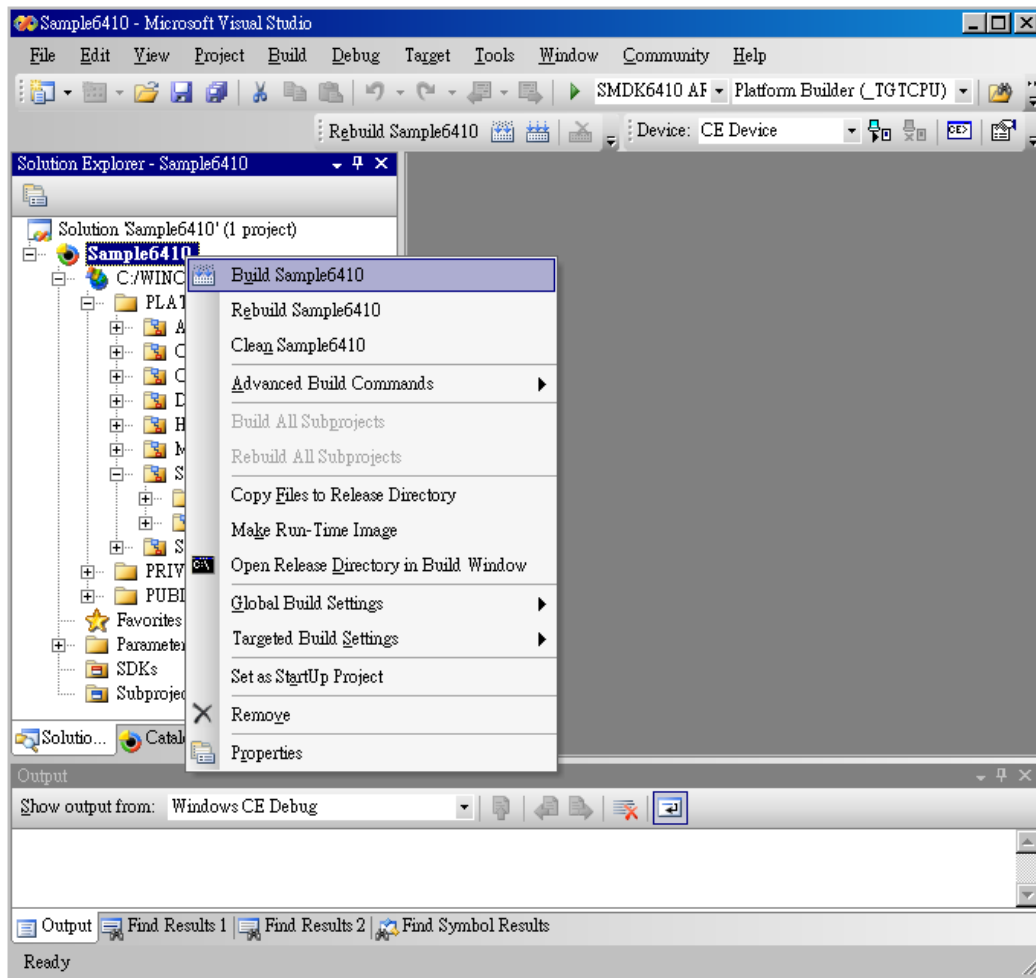
<8> Select “Build Options”, uncheck “Enable kernel debugger”
 , uncheck “Enable KITL”
 , check “Run-time image can be larger than 32MB”.



- <9> Select "Sample6410", press mouse right button,
Select "Target Build Settings",
and uncheck "Make Run-Time Image After Building"



<10> Select “Sample6410”, press mouse right button, Select “Build Sample6410”



3.3 Install BSP SMDKC100

This is for HMI 750, 1050 & 1550 only

Quick Procedure

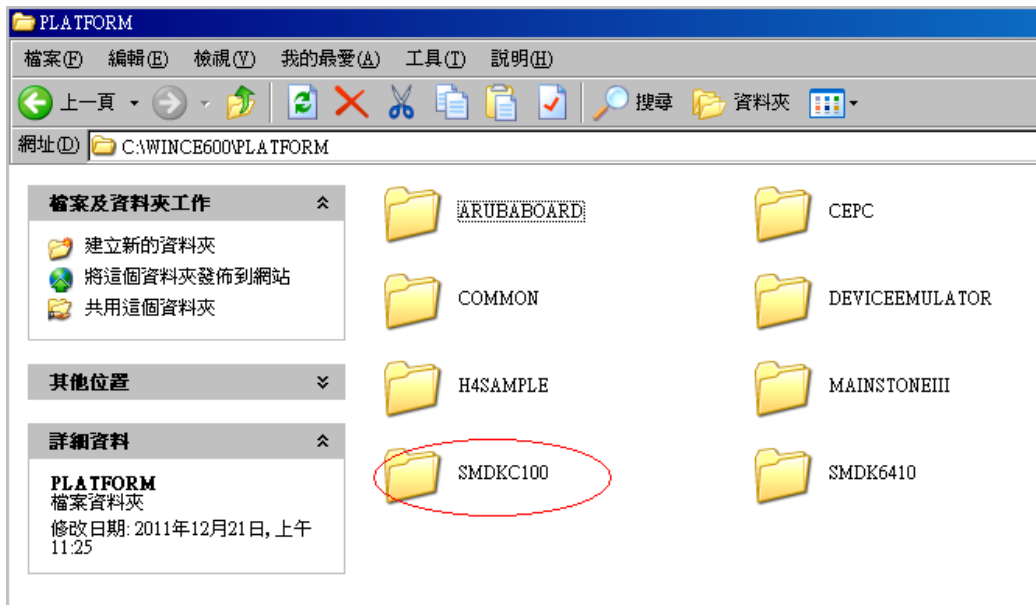
1. Decompress the BSP compression file.(SMDKC100_1V2B11.rar)
2. Copy folder “SMDKC100” to “C:\WINCE600\PLATFORM”.
3. Under “C:\WINCE600\OSDesigns” create a project folder.
For example: “C:\WINCE600\OSDesigns\MyDevice”.
4. Copy “SamplePC100.pbxml” to folder “C:\WINCE600\OSDesigns\MyDevice”.
5. Double click “SamplePC100.pbxml”.
6. Change build mode : “SMDKC100 ARMV4I Release”
7. Select “SamplePC100”, press mouse right button, select “properties”.
Select “Build Options”, uncheck “Enable kernel debugger”, uncheck “Enable KITL”, check “Run-time image can be larger than 32MB”.
8. Select “SamplePC100”, press mouse right button, select “Build SamplePC100”.

Detailed Procedure

<1> Decompress the BSP compression file.(SMDKC100_1V2B11.rar)

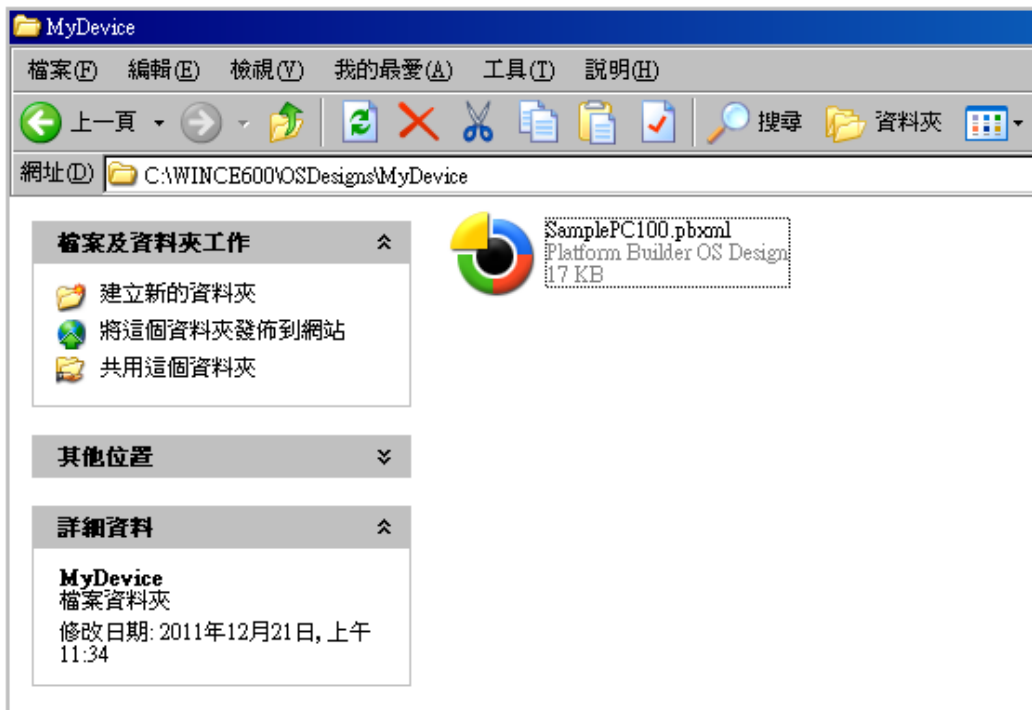


<2> Copy folder “SMDKC100” to “C:\WINCE600\PLATFORM”.



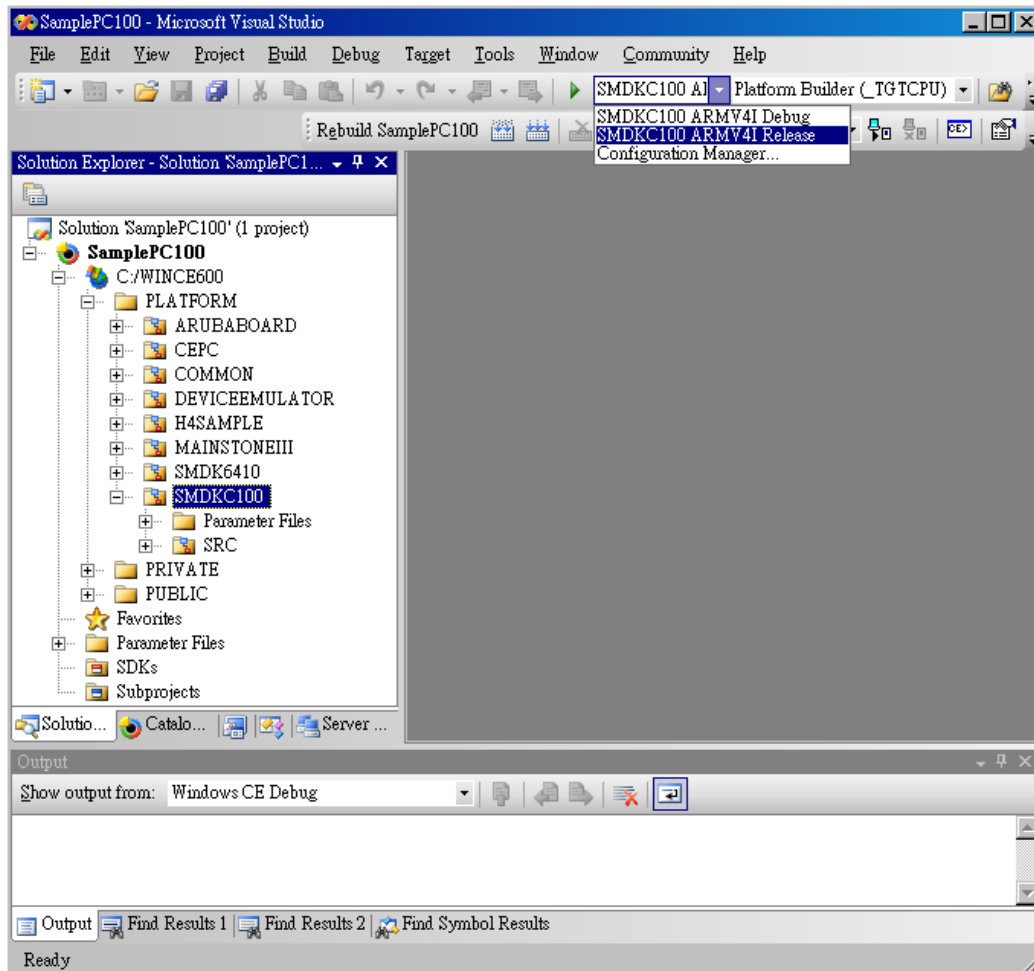
<3> Under “C:\WINCE600\OSDesigns” create a project folder.
For example: “C:\WINCE600\OSDesigns\MyDevice”.

<4> Copy “SamplePC100.pbxml” to folder
“C:\WINCE600\OSDesigns\MyDevice”.

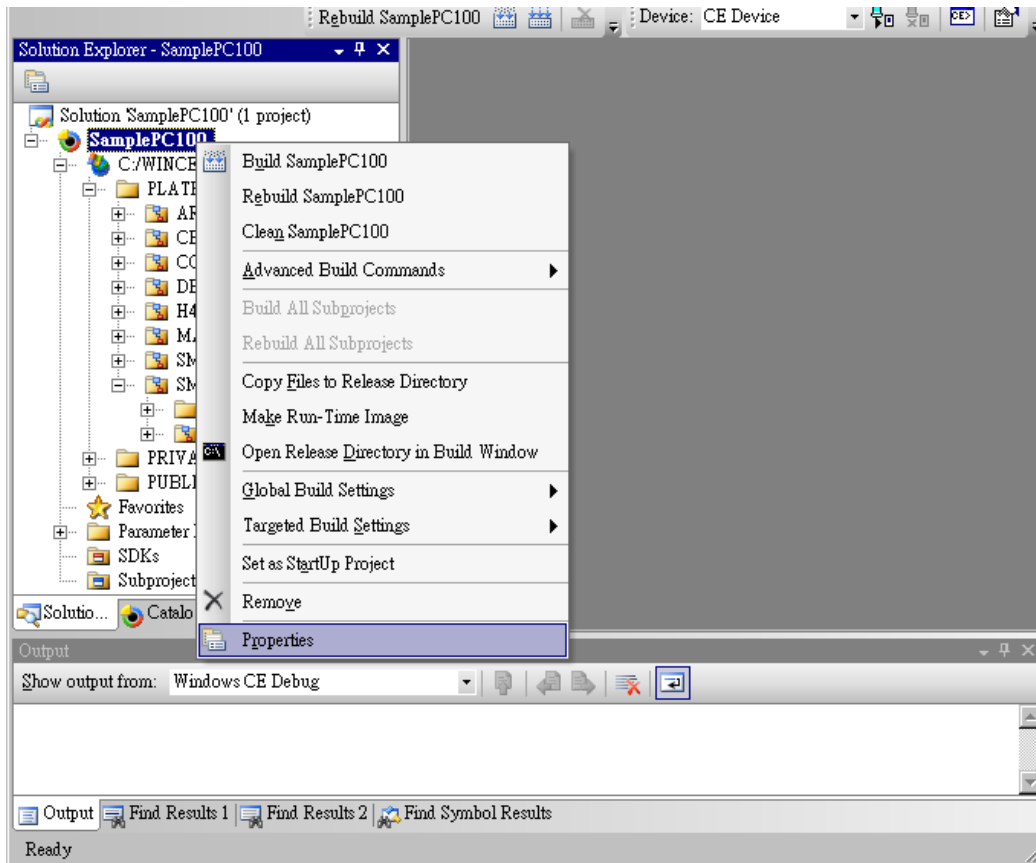


<5> Double click “SamplePC100.pbxml”.

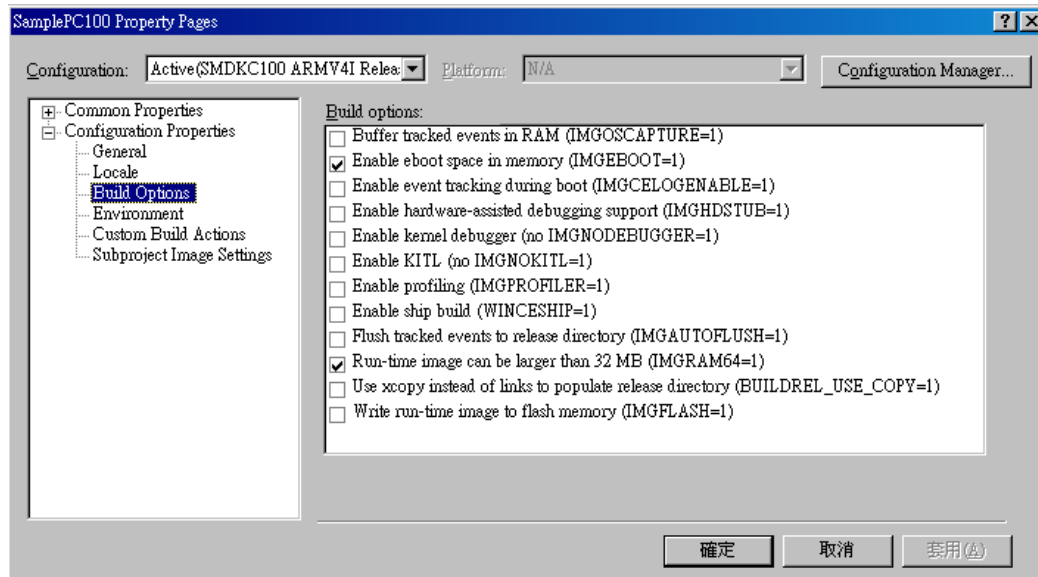
<6> Change build mode : “SMDKC100 ARMV4I Release”



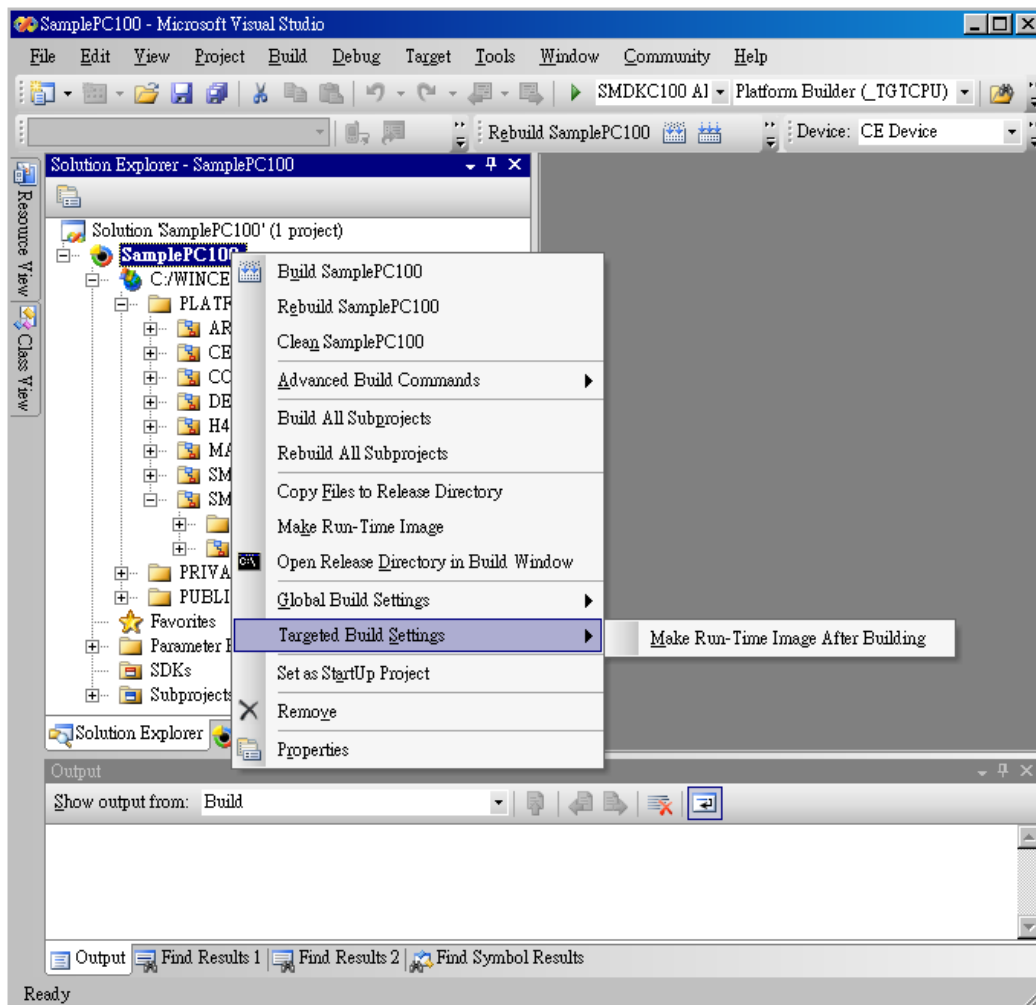
<7> Select "SamplePC100", press mouse right button, select "properties".



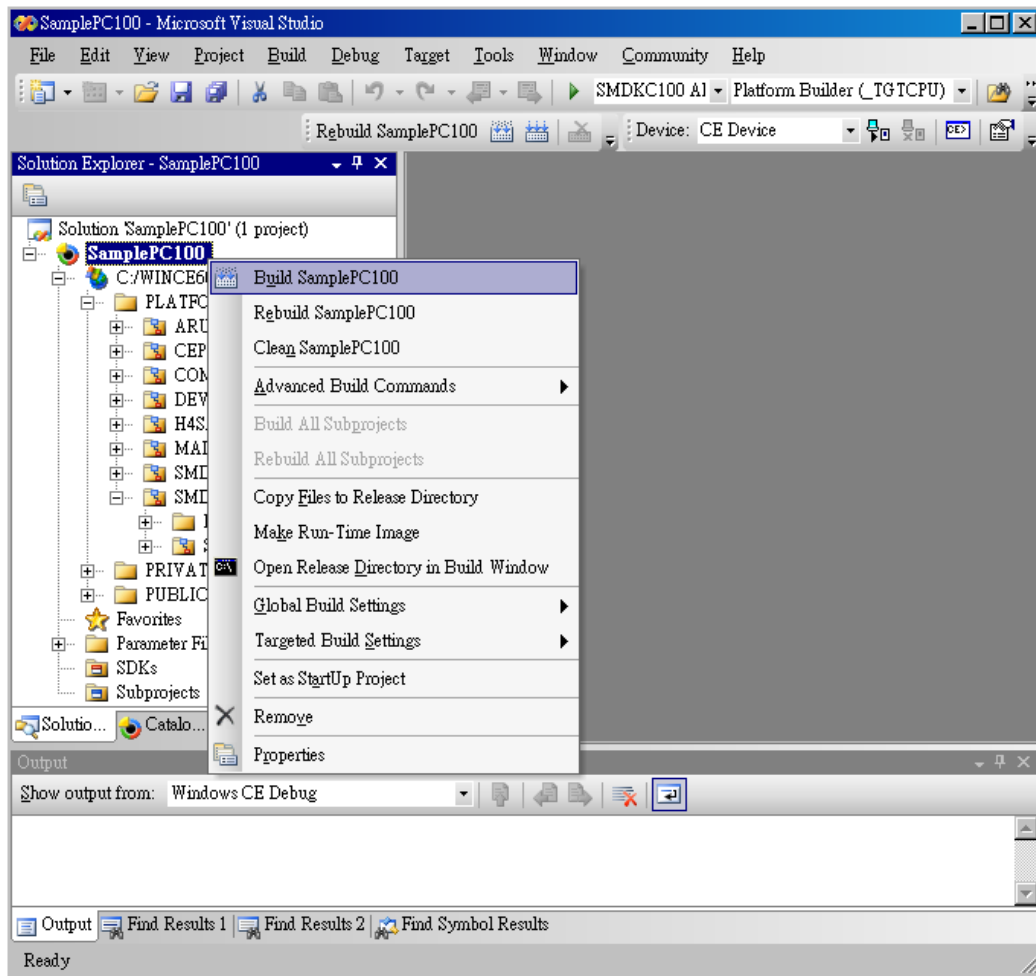
<8> Select “Build Options”, uncheck “Enable kernel debugger”
, uncheck “Enable KITL”
, check “Run-time image can be larger than 32MB”.



<9> Select "SamplePC100", press mouse right button,
Select "Target Build Settings", and uncheck "Make Run-Time Image After Building"



<10> Select “SamplePC100”, press mouse right button, Select “Build SamplePC100”



3.4 Add “User Account” into Project

Quick Procedure

1. Decompress “UserAccount.rar”
2. Copy folder “UserAccount” to your project folder.
For example : “C:\WINCE600\OSDesigns\MyDevice\UserAccount”.
3. Open your project: Select “Subprojects”, press mouse right button, select “Add Existing Subproject”.

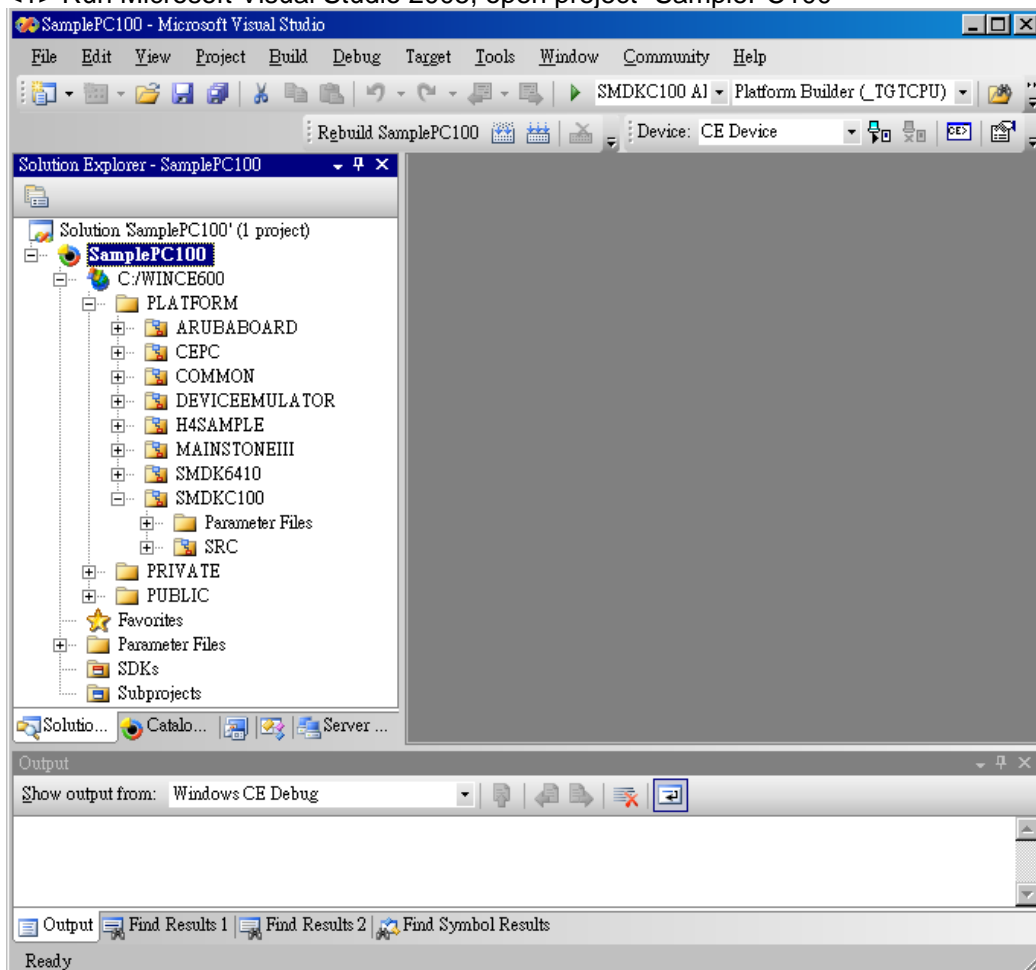
Find the file

“C:\WINCE600\OSDesigns\MyDevice\UserAccount\UserAccount.pbpxml” and open it.

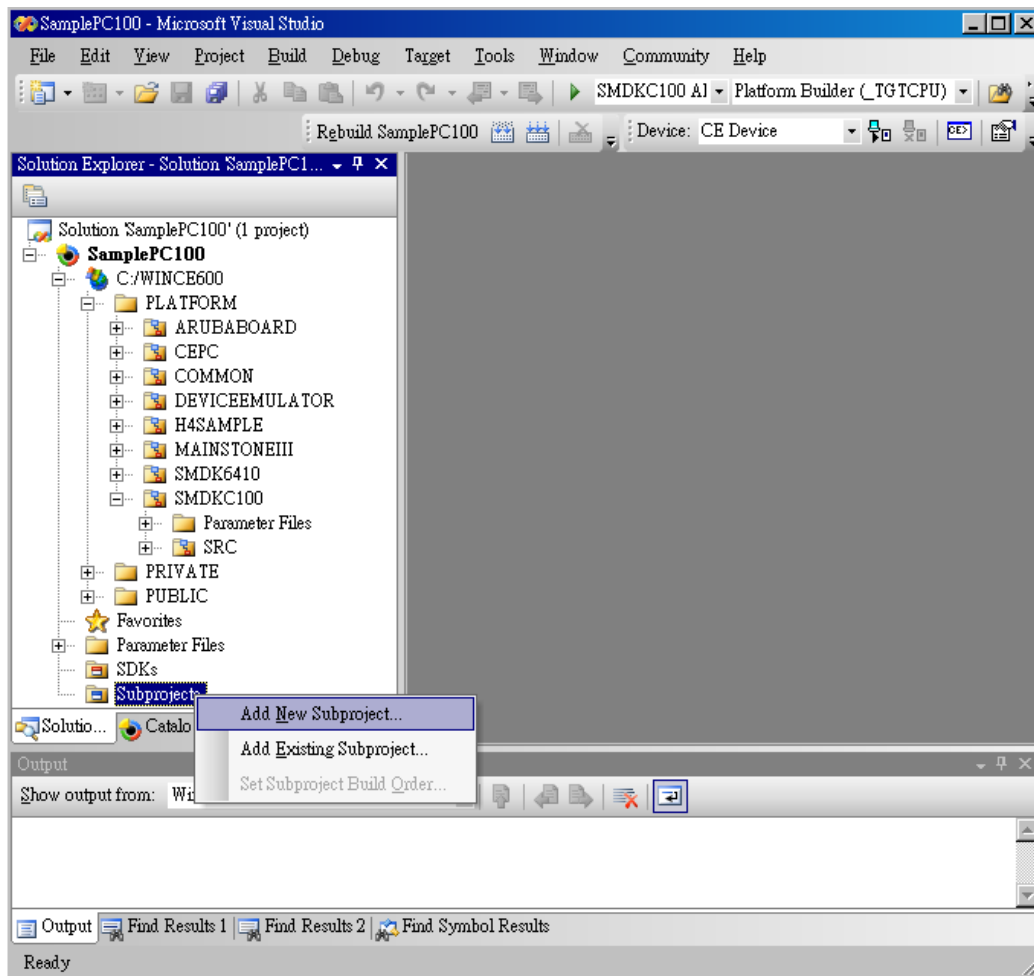
4. Under “Sunprojects”, select “UserAccount”, press mouse right button, select “Build”.

Detailed Procedure

<1> Run Microsoft Visual Studio 2005, open project “SamplePC100”




<2> Select “Subprojects”, press mouse right button, select “Add New Subproject...”



<3> Press “Next”

Windows Embedded CE Subproject Wizard

 **Select name, location and template**

Available templates:

- WCE Application
- WCE Console Application**
- WCE Dynamic-Link Library
- WCE Static Library
- WCE TUX Dynamic-Link Library

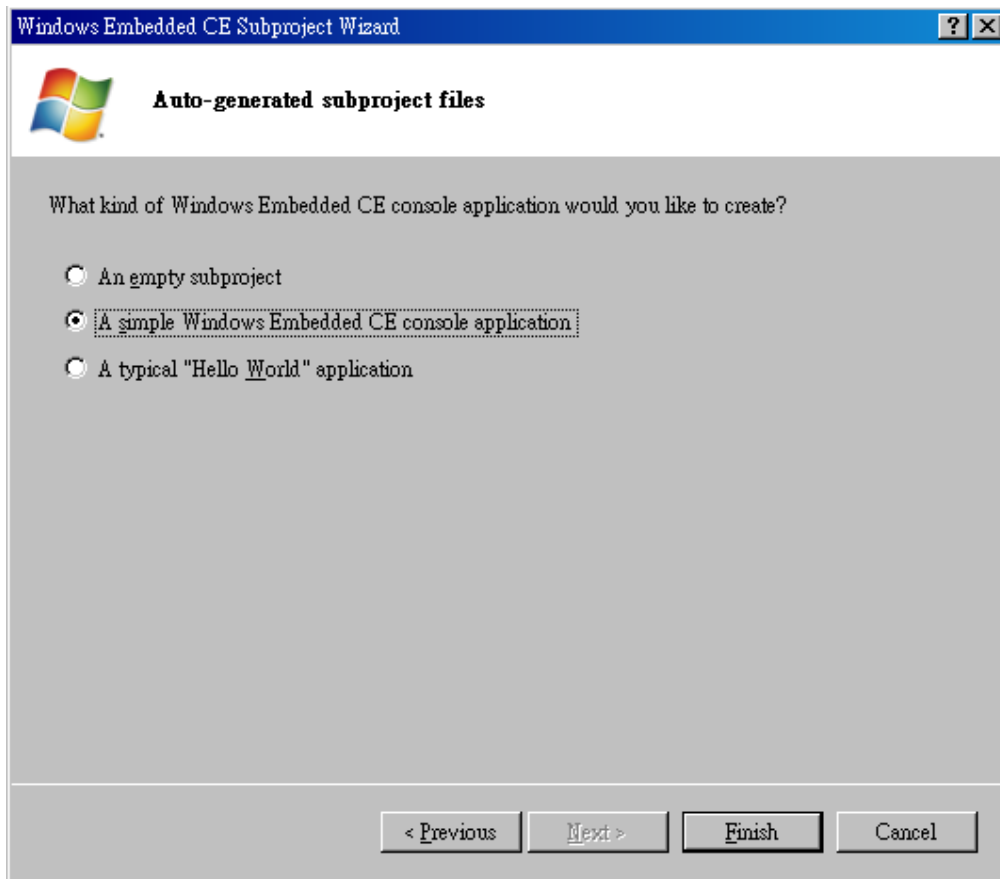
Subproject name:

Location:

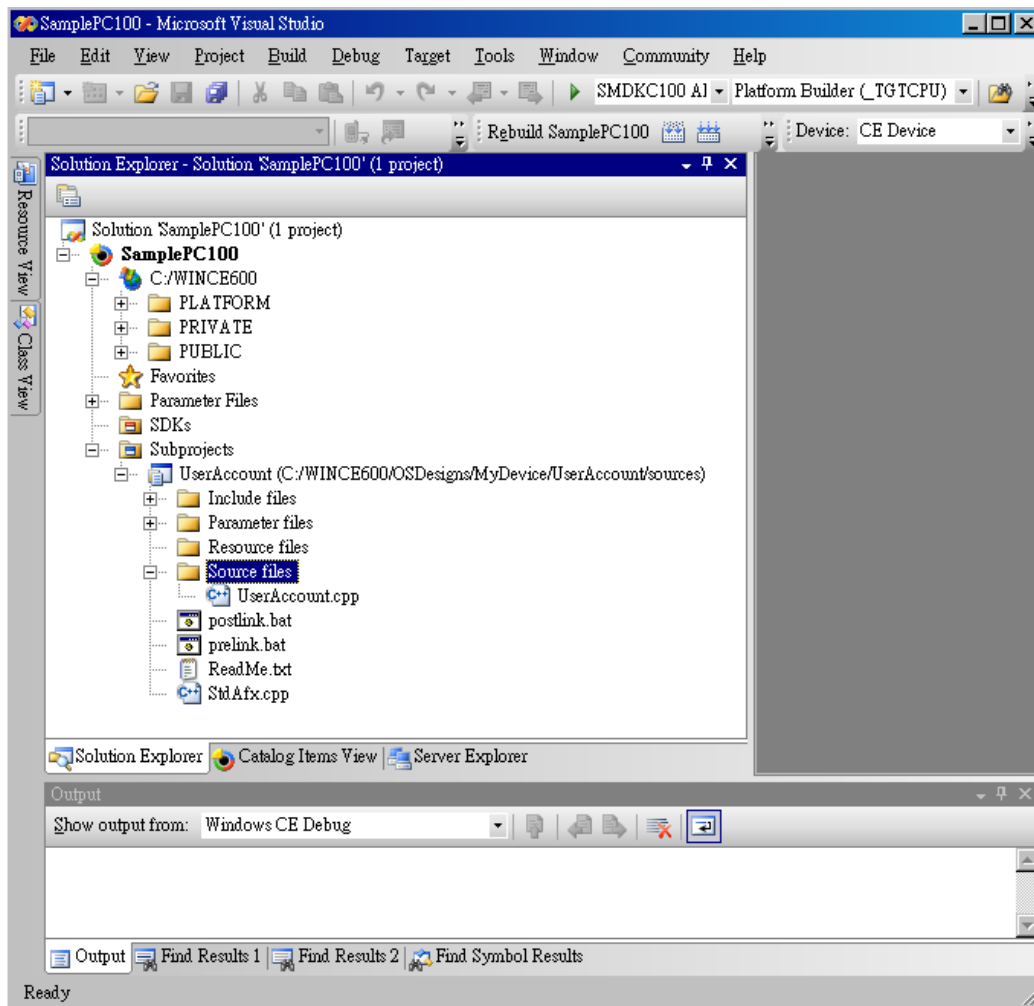
< Previous Next > Finish Cancel

<4> Select "A simple Windows Embedded CE console application".

Press "Finish"



<5> Expand “SamplePC100-Subprojects-UserAccount-Source files”



<6> Open file

"C:\WINCE600\OSDesigns\MyDevice\UserAccount\UserAccount.cpp"

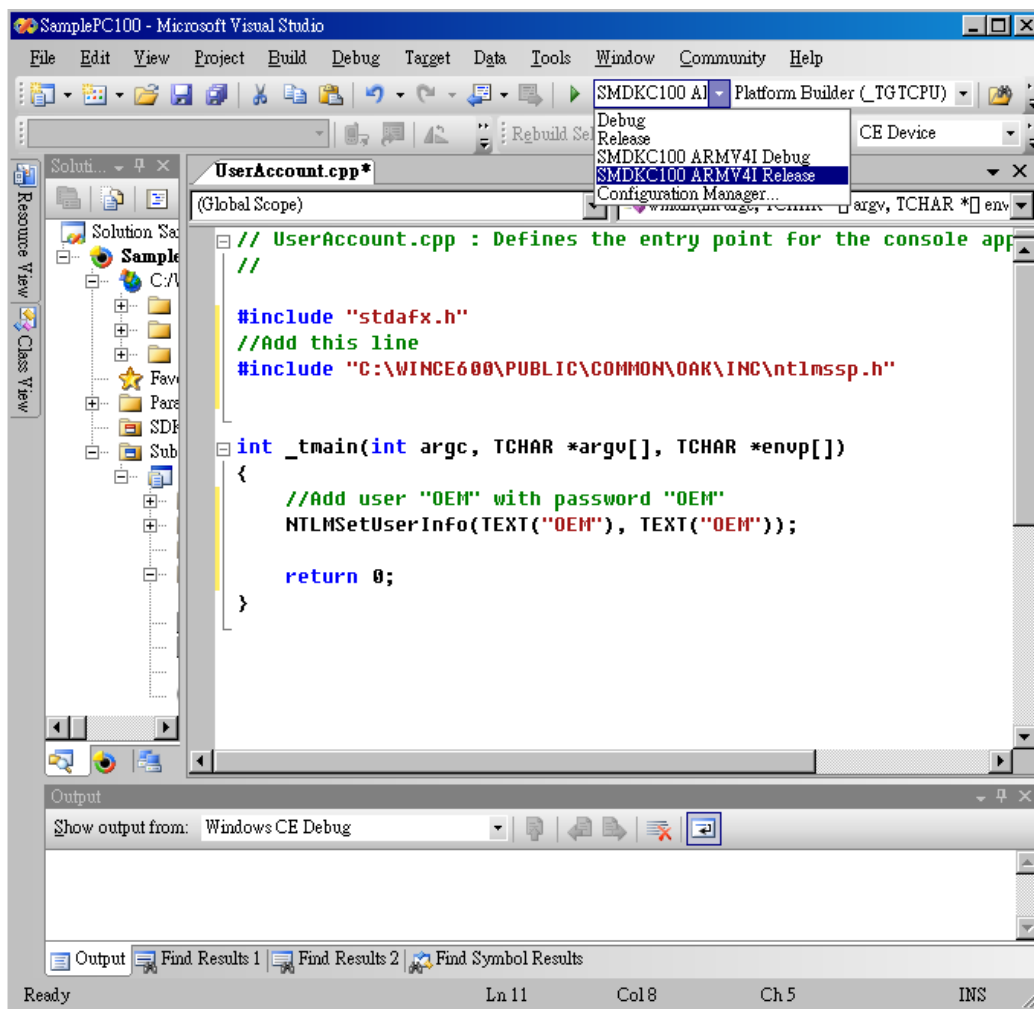
Add include line

```
#include "C:\WINCE600\PUBLIC\COMMON\OAK\INC\ntlmssp.h"
```

Add code lines in _tmain()

```
//Add user "OEM" with password "OEM"
```

```
NTLMSetUserInfo(TEXT("OEM"), TEXT("OEM"));
```



<7> Open file

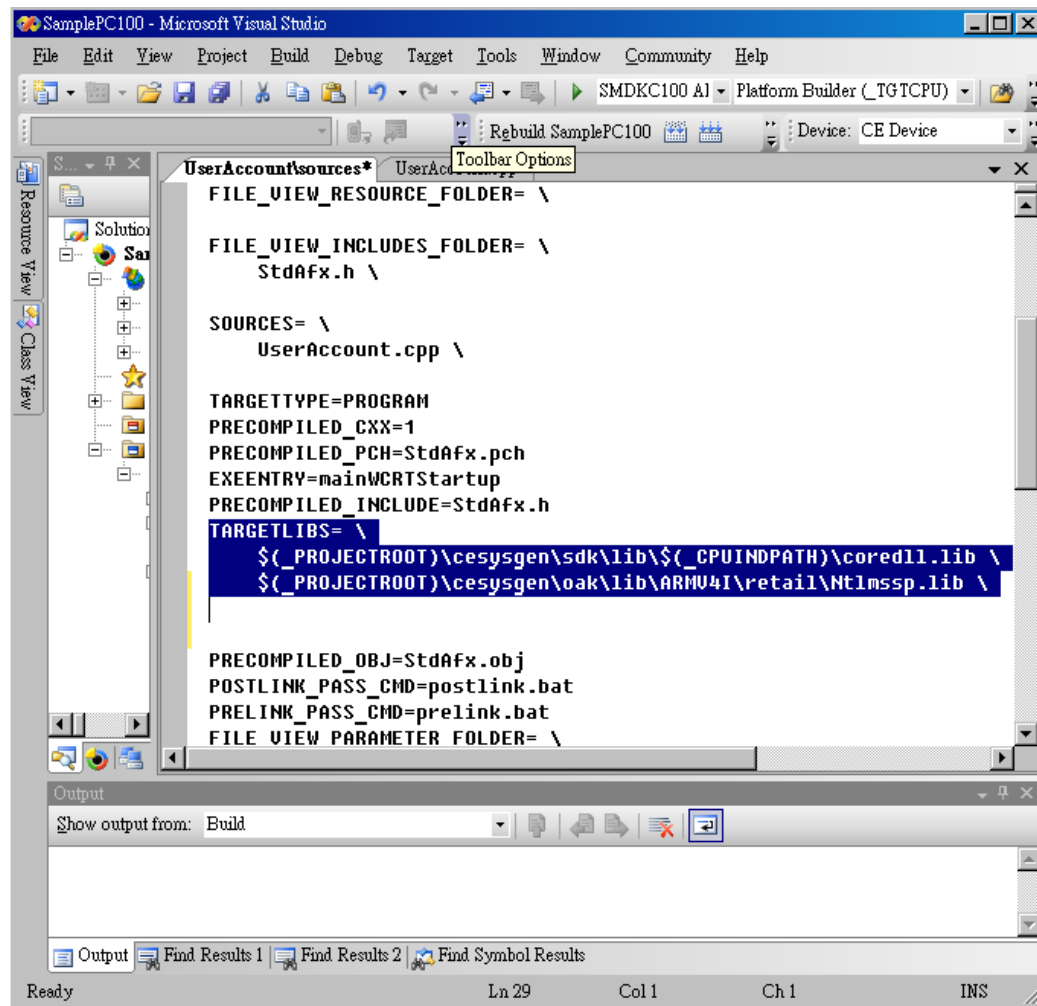
"C:\WINCE600\OSDesigns\MyDevice\UserAccount\sources"

Add a line to append "TARGETLIBS":

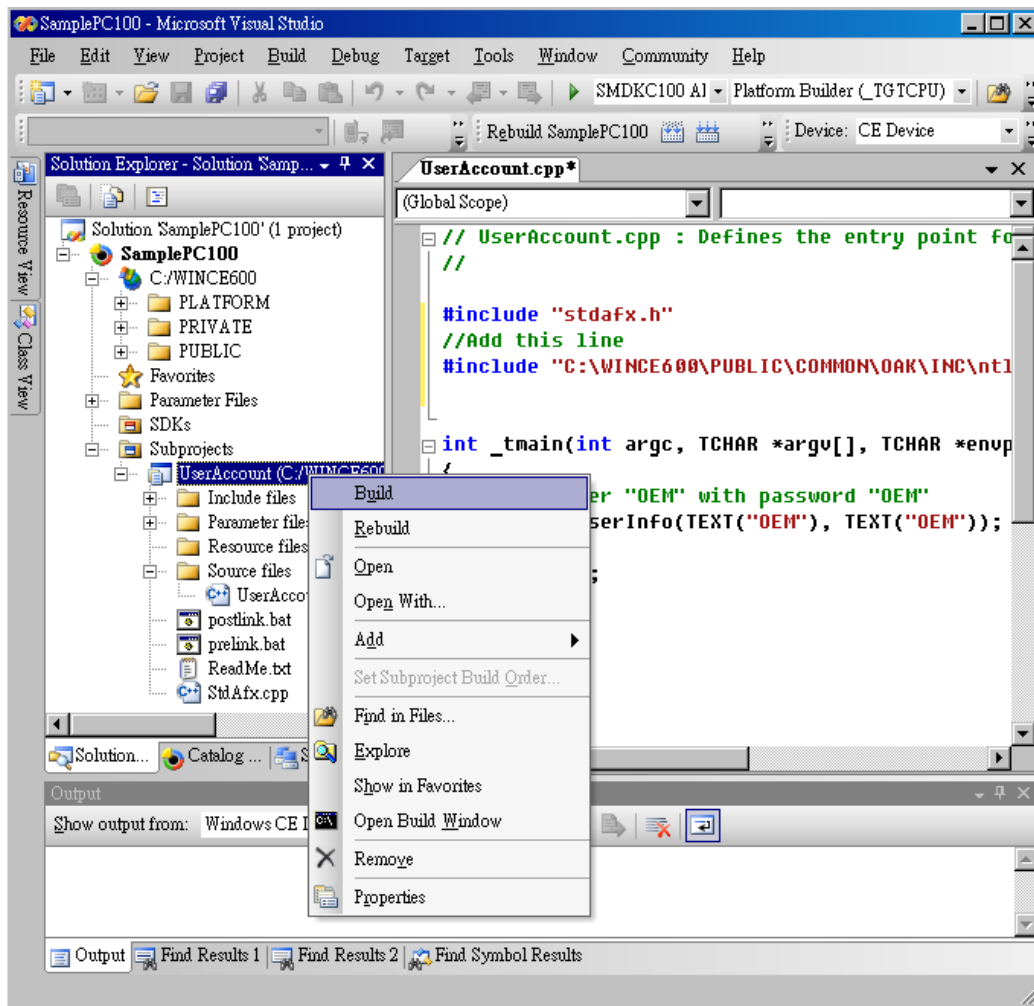
TARGETLIBS= \

\$(_PROJECTROOT)\cesysgen\sdk\lib\\$(_CPUINDPATH)\coredll.lib \

\$(_PROJECTROOT)\cesysgen\oak\lib\ARMV4I\retail\Ntlmssp.lib \



<8> Select "UserAccount", press mouse right button, select "Build".



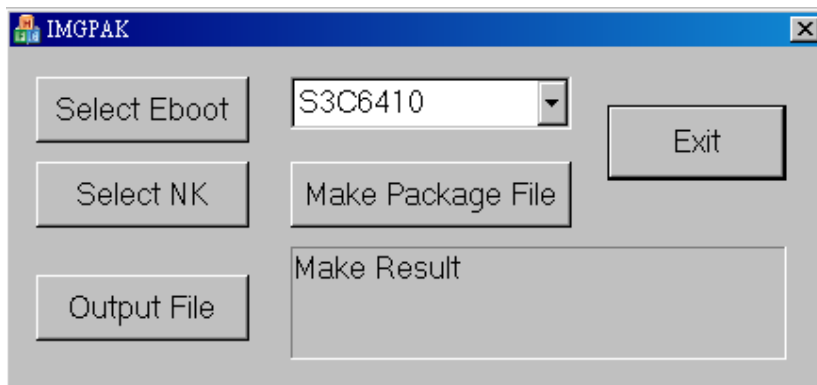
3.5 Use “IMGPAK.EXE” to build OEM_6410.Bin (OEM_PC100.Bin)

Quick Procedure

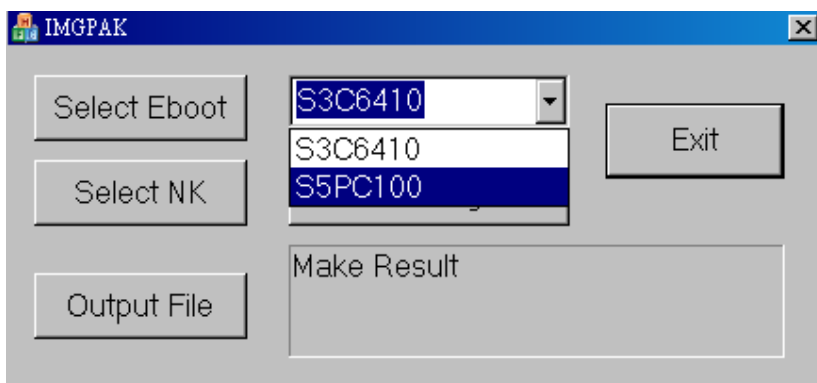
1. Run IMGPAK.exe on your PC.
2. Select CPU type.(S3C6410 or S5PC100)
3. Push button “Select Eboot” to select the Eboot.bin that you want to package.
4. Push button “Select NK” to select the Nk.bin that you want to package.
5. Push button “Output File” to select the output package file.
6. Push button “Make Package File” to make package file.

Detailed Procedure

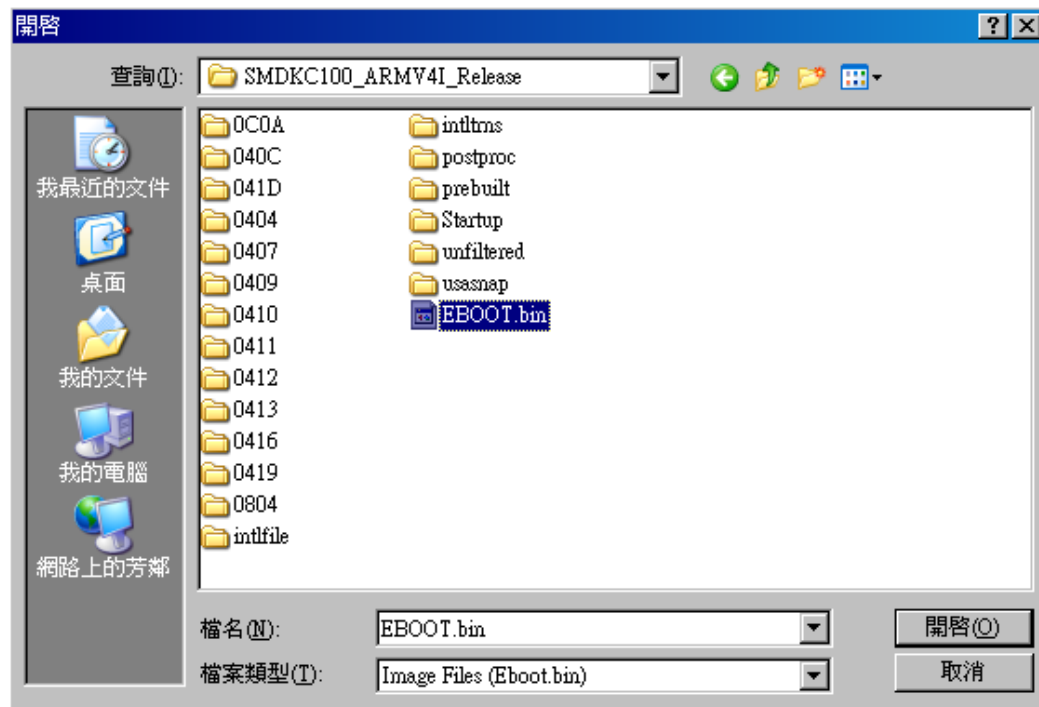
<1> Run IMGPAK.exe on your PC.



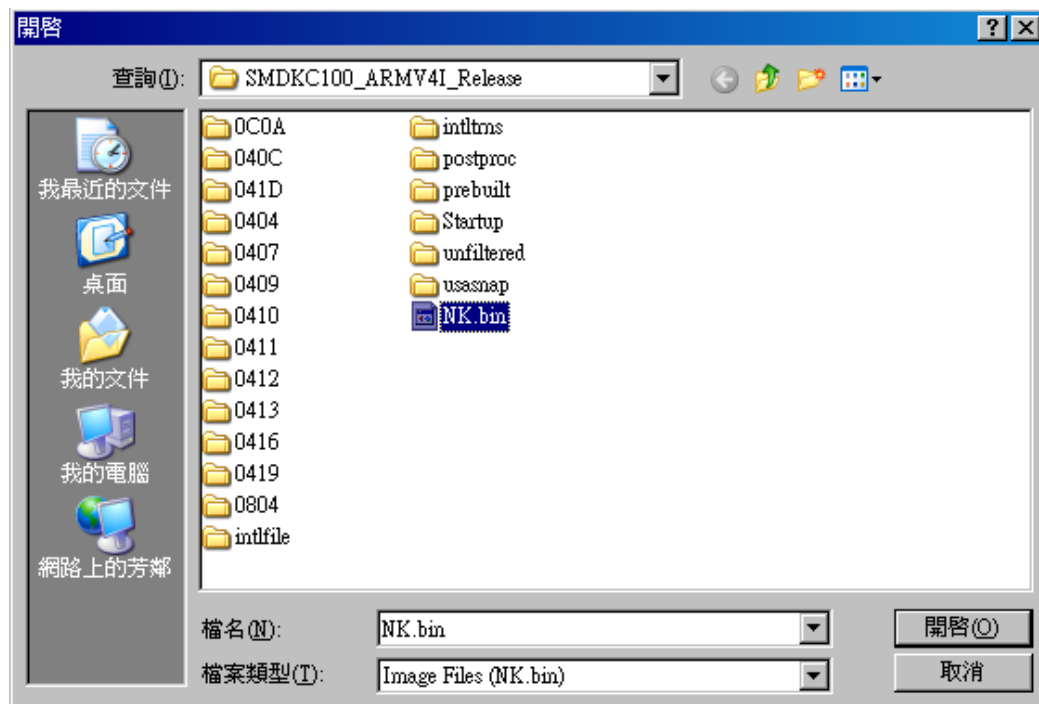
<2> Select CPU type.(S3C6410 or S5PC100)



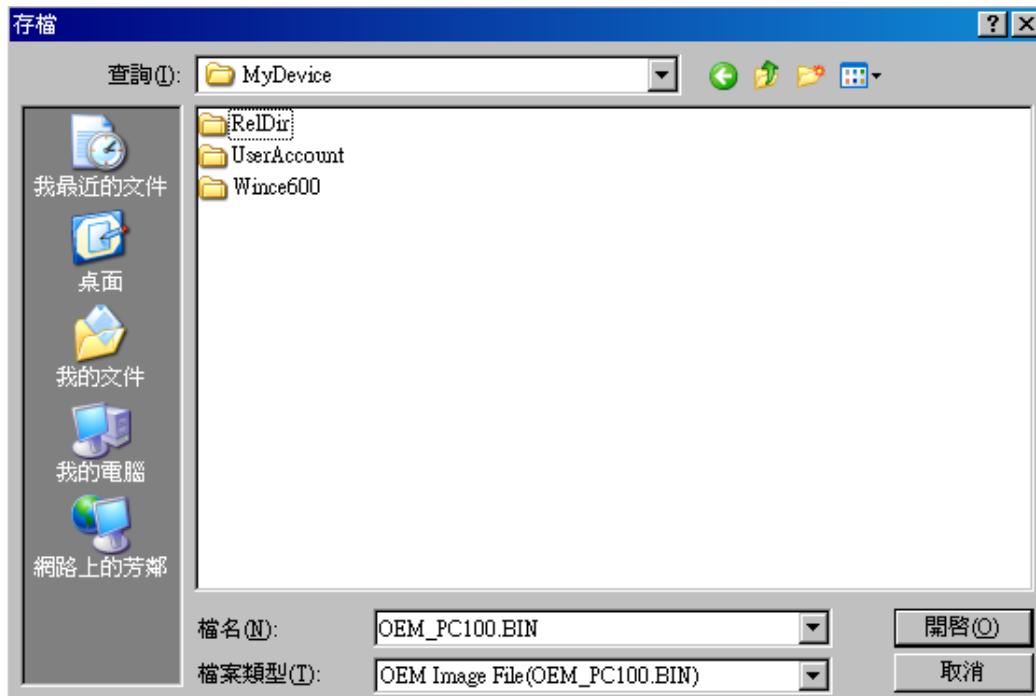
<3> Press button “Select Eboot” to select the Eboot.bin that you want to package. Press “Open”.



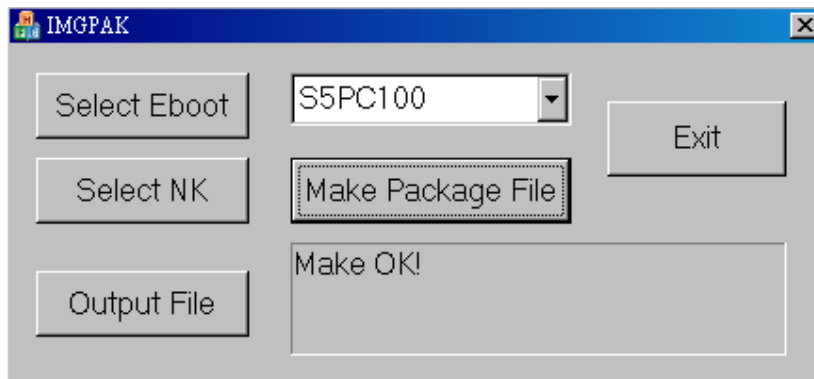
<4> Press button “Select NK” to select the Nk.bin that you want to package. Press “Open”.



<5> Press button “Output File” to select the output package file.
Press “Open”.



<6> Press button “Make Package File” to make package file.



3.6 How to Change the Boot Logo

<1> Prepare your jpg logo file.

Image resolution

HMI size	Resolution
HMI 4.3"	480x272
HMI 7"	800x480(7")
HMI 10"/15"	1024x768

The device decoder only support "Sequential Encoding" compression mode.
If jpg image is too big, the eboot.bin will build fail.

Please use the windows tool "Paint.exe" to save as a standard jpg format file.

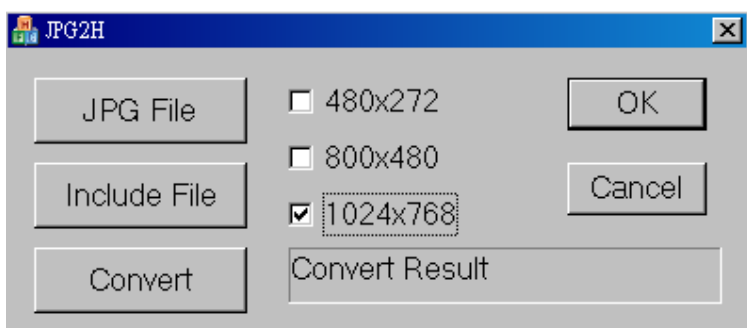
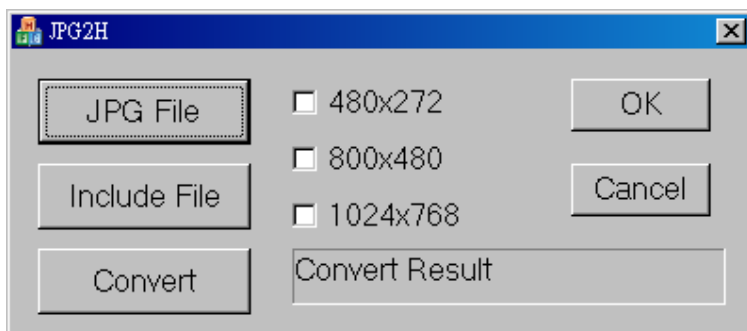
<2> Run "JPG2H.exe".

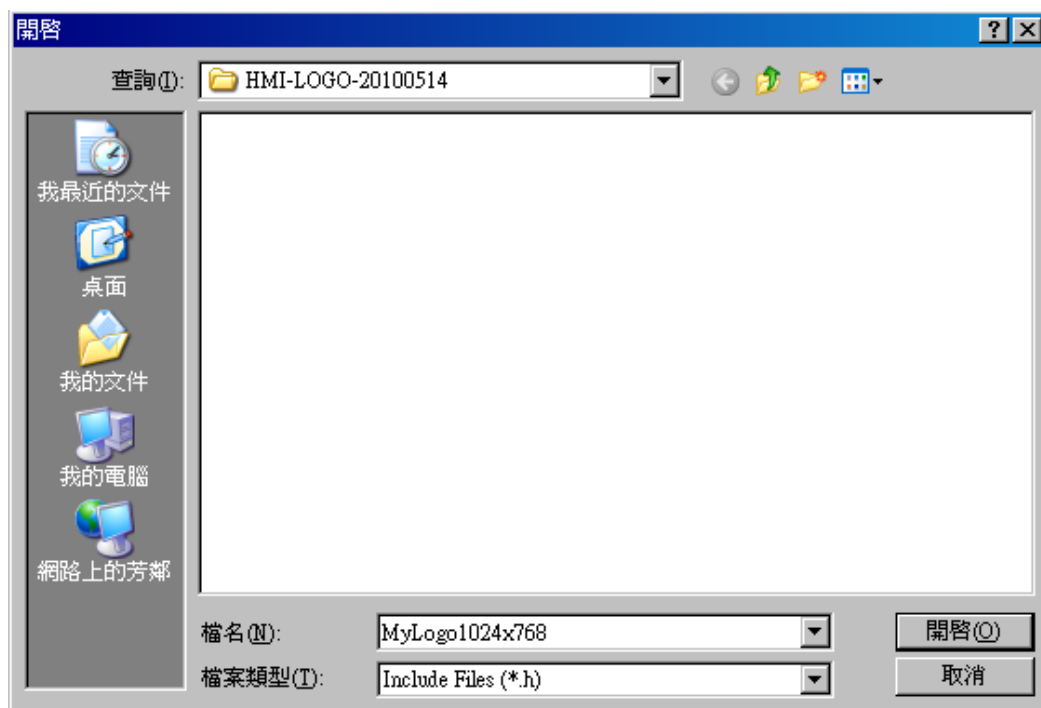
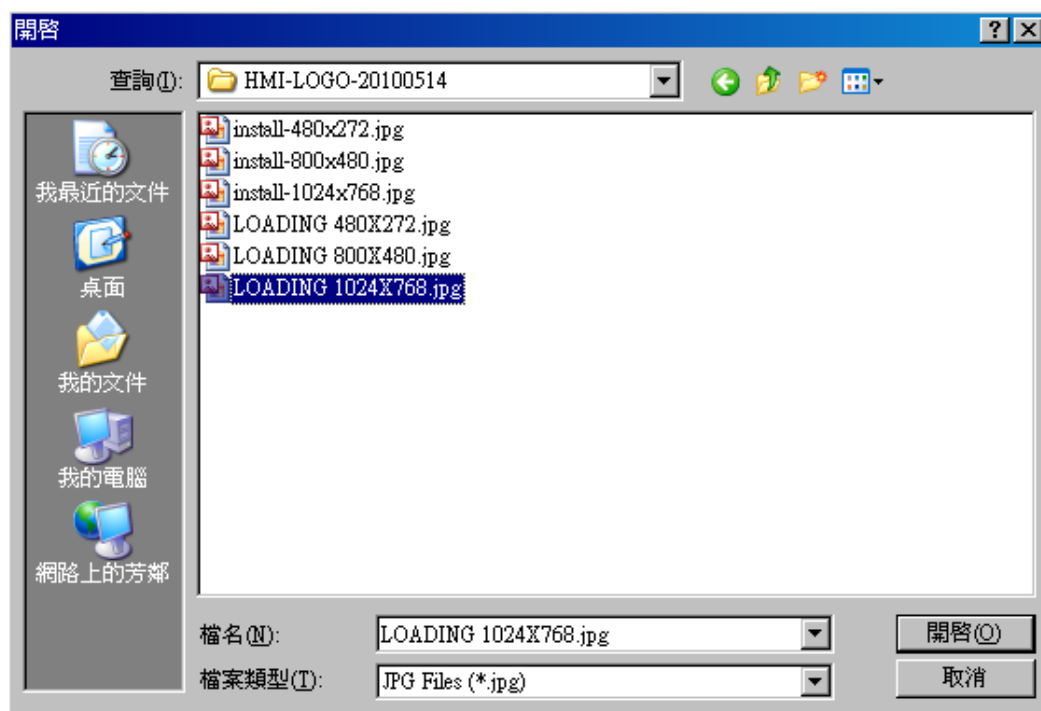
Select the resolution of your jpg file.

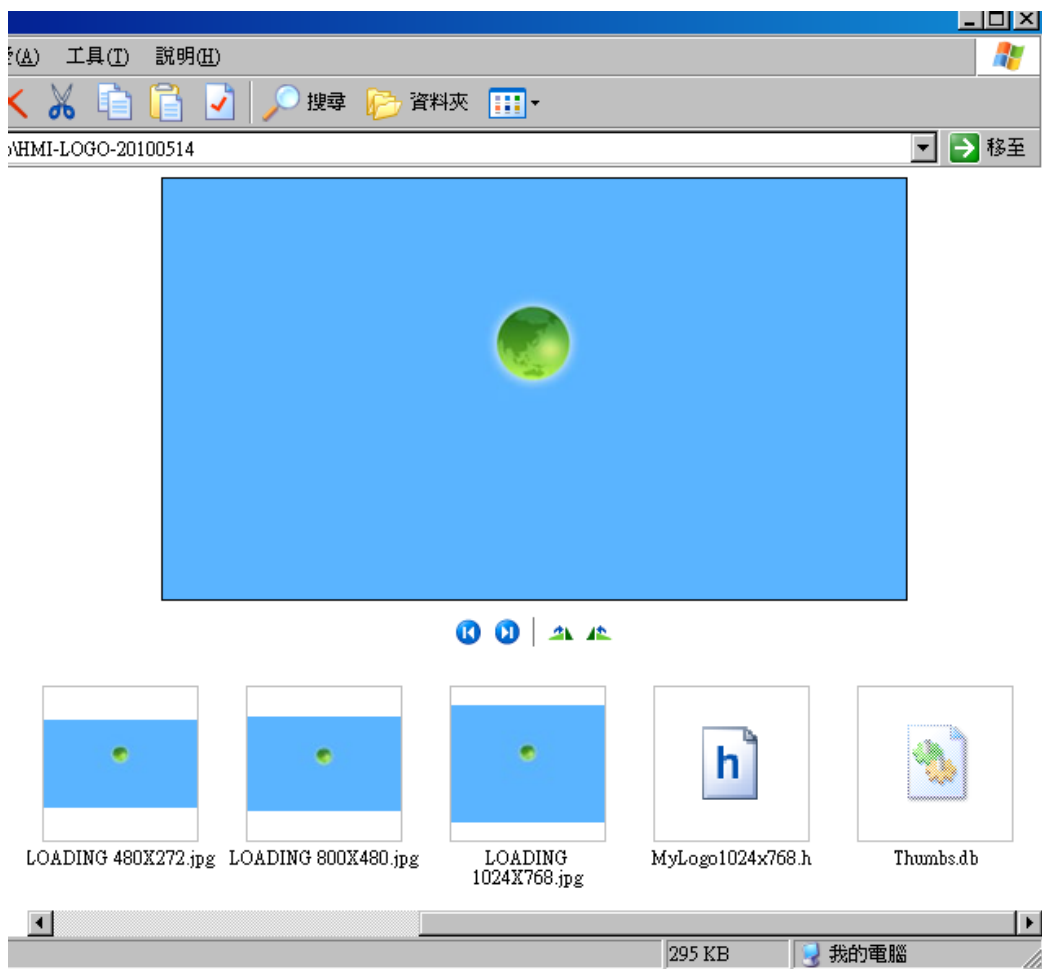
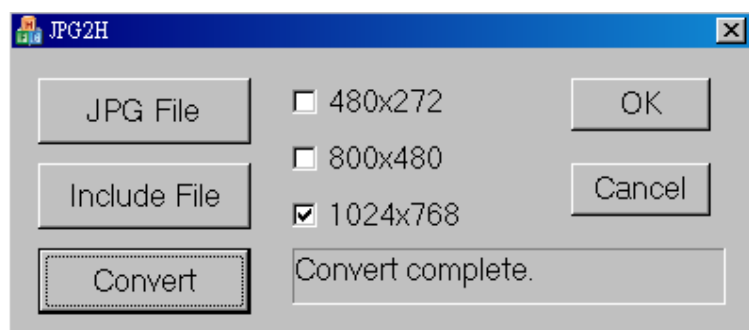
Click "JPG File" to open the .jpg file.

Click "Include File" to open the output .h file

Click "Convert" to convert from jpg file to h file.



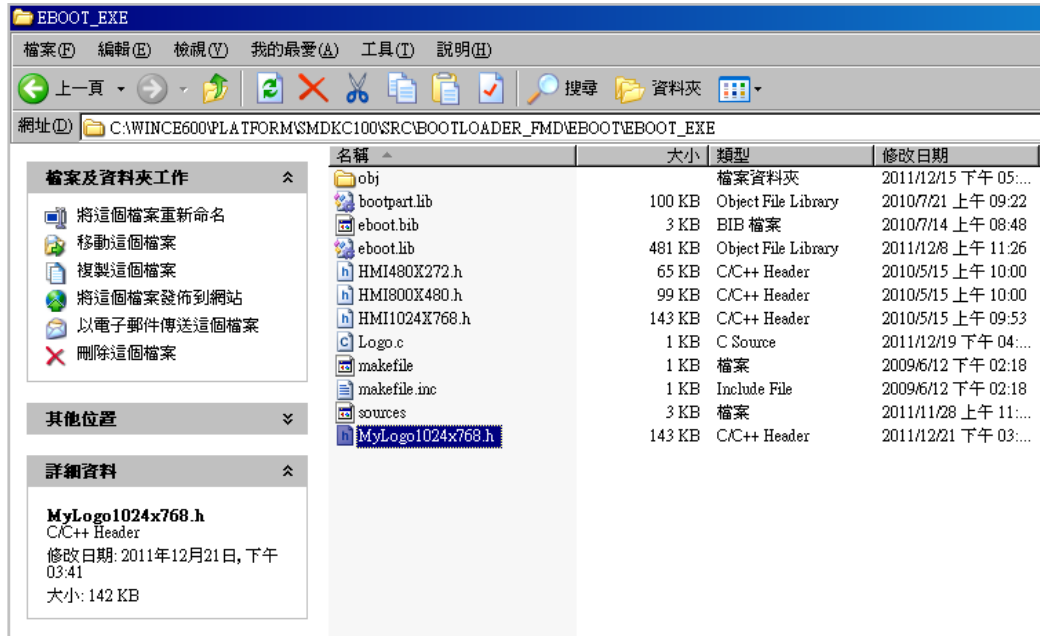




<3> Copy file “MyLogo1024x768 .h” to the BSP folder:

For example,

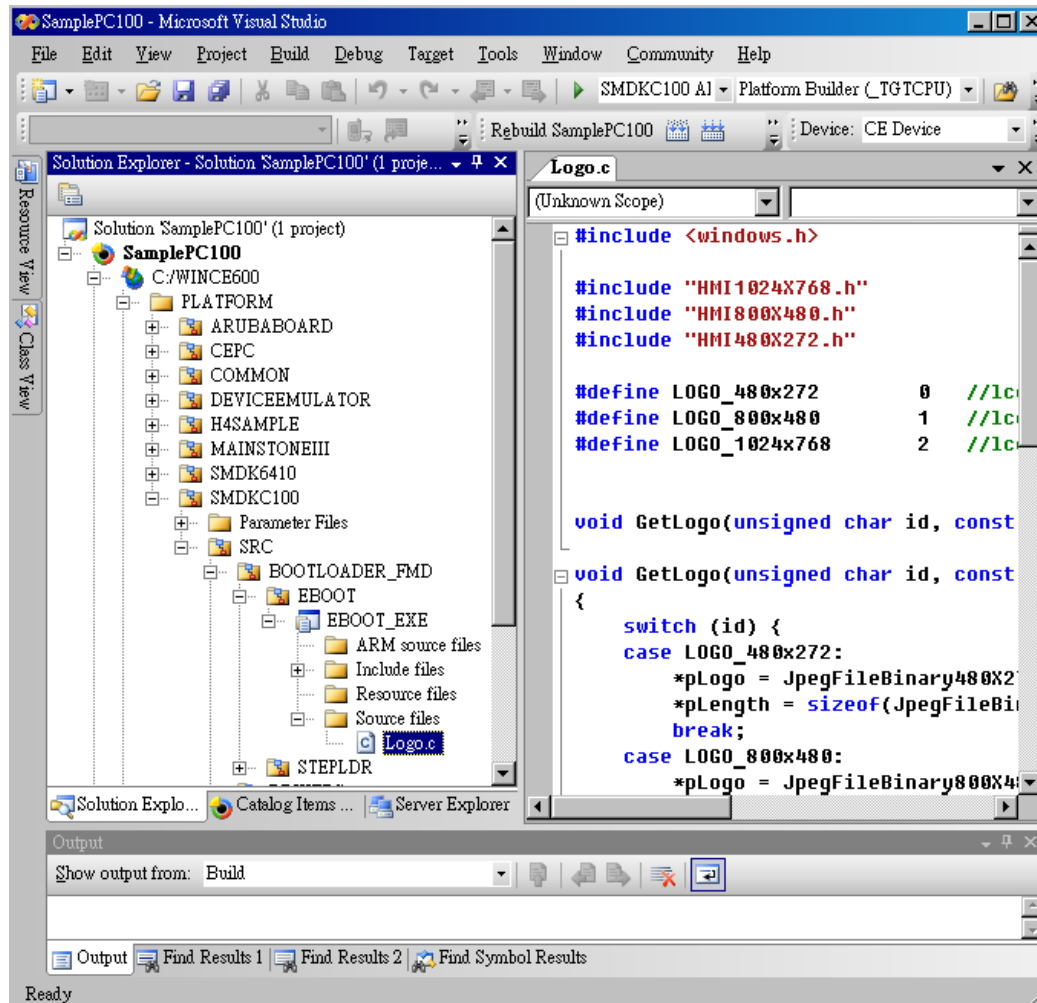
C:\WINCE600\PLATFORM\SMDKPC100\SRC\BOOTLOADER\EBOOT\EBOOT_EXE



<4> In the platform build,

Expend “SamplePC100-C:/WINCE600-PLATFORM-SMDKC100-SRC-
BOOTLOADER_FMD-EBOOT-EBOOT_EXE-Source files”

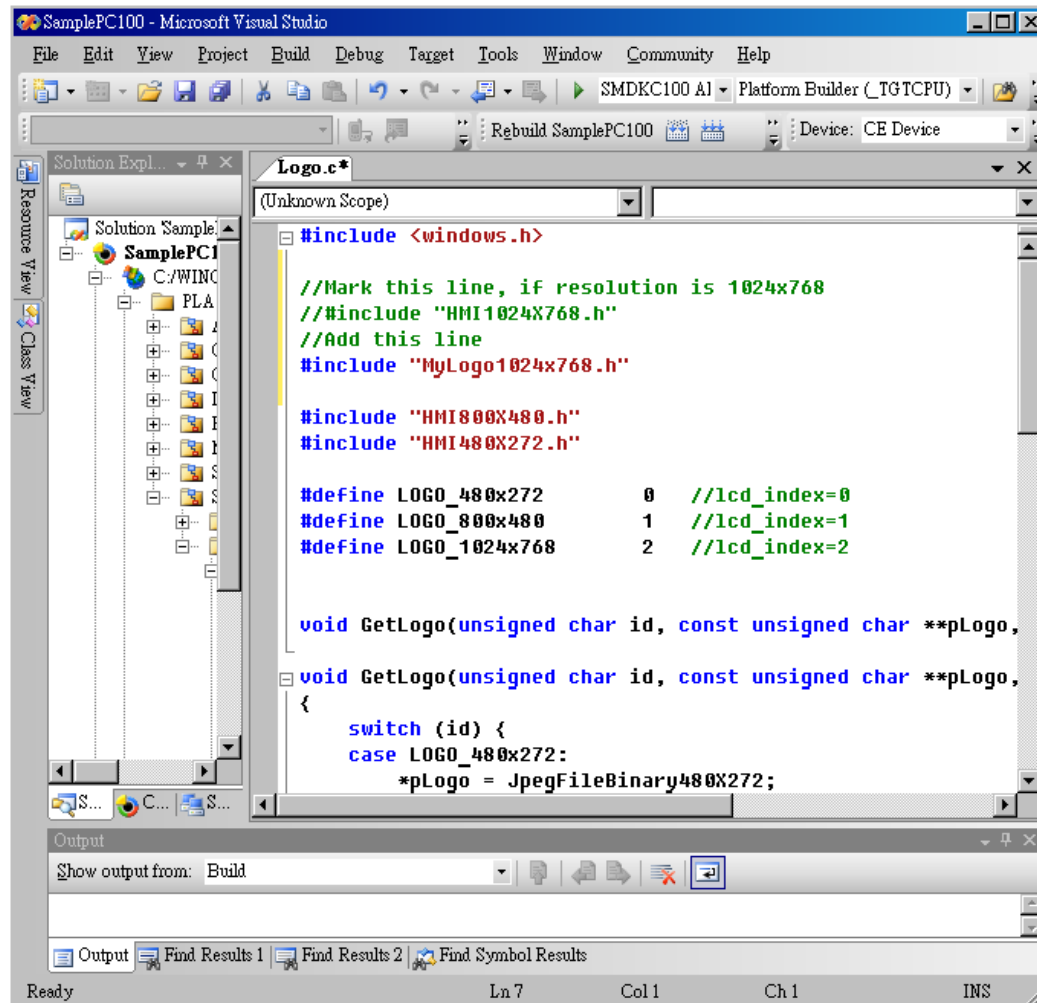
Open “Logo.c”



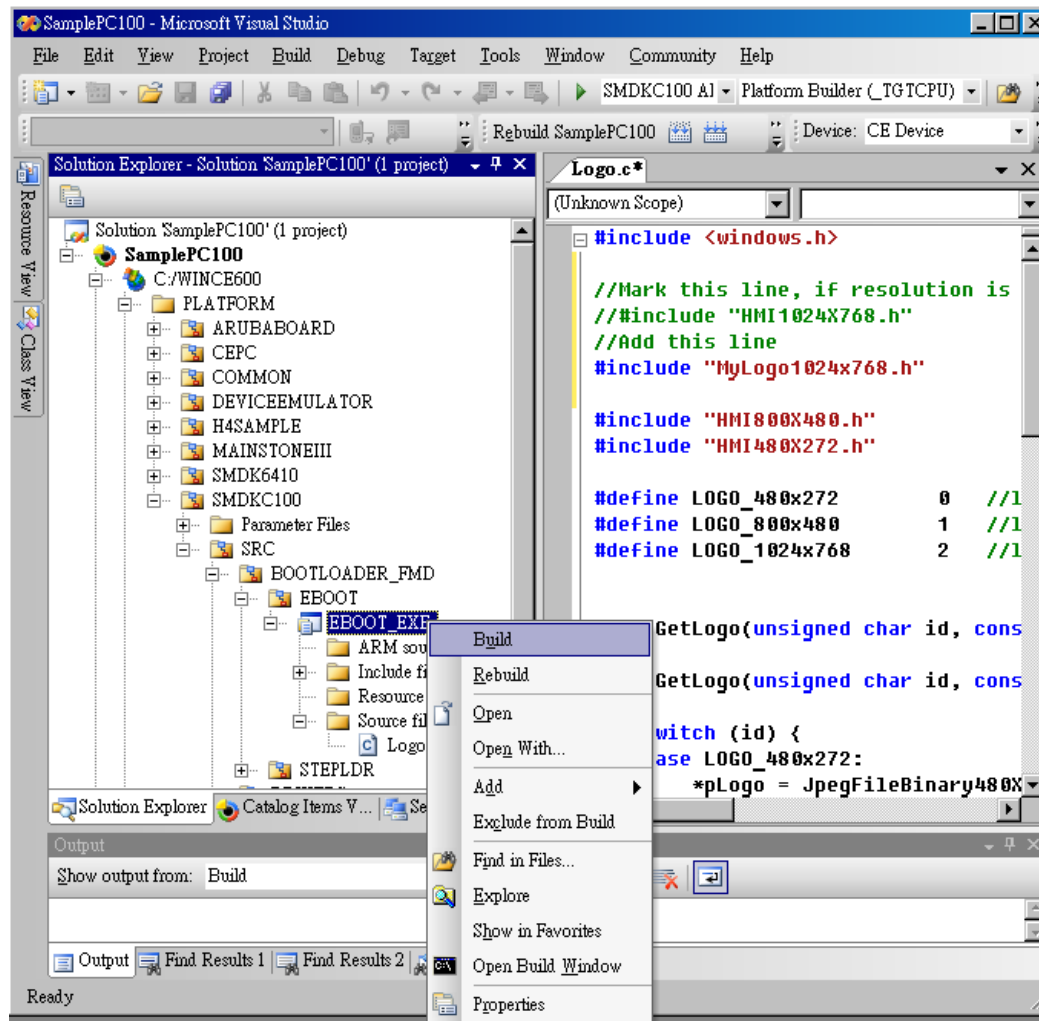
<5> Mark the line "#include HMI1024X768.h", and add include your MyLogo1024x768.h file.

```
//Mark this line, if resolution is 1024x768  
//#include "HMI1024X768.h"  
//Add this line
```

```
#include "MyLogo1024x768.h"
```



<6> In the platform build, select “eboot_exe”, press mouse right button, select “Build”.



3.7 Buzzer Control

<1> Sample Code

```
//IOAPI.dll function define
typedef BOOL FUNC_IOAPI_Init();
typedef void FUNC_IOAPI_Deinit();
typedef BOOL FUNC_BuzzerSetVolume(unsigned int vol);
typedef BOOL FUNC_BuzzerGetVolume(unsigned int *pvol);
typedef BOOL FUNC_BuzzerSetOnOff(BOOL onoff);

//Declare function pointers
FUNC_IOAPI_Init          *pfnIOAPI_Init = NULL;
FUNC_IOAPI_Deinit        *pfnIOAPI_Deinit = NULL;
FUNC_BuzzerSetVolume      *pfnBuzzerSetVolume = NULL;
FUNC_BuzzerGetVolume      *pfnBuzzerGetVolume = NULL;
FUNC_BuzzerSetOnOff       *pfnBuzzerSetOnOff = NULL;

//Declare dll handle
HMODULE hIOAPI = NULL;

void IOAPITEST_Deinit()
{
    FreeLibrary(hIOAPI);
    hIOAPI = NULL;
}

BOOL IOAPITEST_Init()
{
    unsigned int rd = 0;

    hIOAPI = LoadLibrary(TEXT("IOAPI.DLL"));
    if (hIOAPI==NULL) {
        RETAILMSG(1, (TEXT("Load IOAPI.dll fail!!!\r\n")));
        return FALSE;
    }
}
```

```

pfnIOAPI_Init = (FUNC_IOAPI_Init *)
GetProcAddress((HMODULE)hIOAPI, TEXT("IOAPI_Init"));
pfnIOAPI_Deinit = (FUNC_IOAPI_Deinit *)
GetProcAddress((HMODULE)hIOAPI, TEXT("IOAPI_Deinit"));
pfnBuzzerSetVolume = (FUNC_BuzzerSetVolume *)
GetProcAddress((HMODULE)hIOAPI, TEXT("BuzzerSetVolume"));
pfnBuzzerGetVolume = (FUNC_BuzzerGetVolume *)
GetProcAddress((HMODULE)hIOAPI, TEXT("BuzzerGetVolume"));
pfnBuzzerSetOnOff = (FUNC_BuzzerSetOnOff *)
GetProcAddress((HMODULE)hIOAPI, TEXT("BuzzerSetOnOff"));

if ( (pfnIOAPI_Init == NULL)
      || (pfnIOAPI_Deinit == NULL)
      || (pfnBuzzerSetVolume == NULL)
      || (pfnBuzzerGetVolume == NULL)
      || (pfnBuzzerSetOnOff == NULL)
    ) {
    RETAILMSG(1, (TEXT("GetProcAddress from IOAPI.dll fail!!!\n")));
    goto IOAPI_Init_exit;
}
return TRUE;

IOAPI_Init_exit:
IOAPITEST_Deinit();
return FALSE;
}

```

```

void TestBuzzer()
{
    unsigned int i, j;

    pfnIOAPI_Init();                //call ioapi initial function
    pfnBuzzerSetVolume(0);          //set volume = 0
    pfnBuzzerSetOnOff(TRUE);        //set buzzer on

    for (i=0; i<=100; i++) {
        pfnBuzzerSetVolume(i);      //set volume = 0..100
        Sleep(100);
    }
    pfnBuzzerSetVolume(0);
    pfnBuzzerSetOnOff(FALSE);       //set buzzer off
    pfnIOAPI_Deinit();              //call ioapi deinitial function
}

int _tmain(int argc, _TCHAR* argv[])
{
    //Load IOAPI.dll
    if (IOAPITEST_Init() == TRUE) {
        TestBuzzer();
    }
    //Free IOAPI.dll
    IOAPITEST_Deinit();

    return 0;
}

```

3.8 Touch Calibration

// useful for OEM using BSP

WinCE supply API

//Touch calibration:
TouchCalibrate();

//System information:

unsigned long MemTotal,MemFreeSpaceRate,MemFreeSpace;
CString FlashPath=_T("\\ResidentFlash");
CString SDPath=_T("\\Storage Card");
CString USBPath=_T("\\Hard Disk");
ULARGE_INTEGER FreeBytes,TotalBytes,TotalFree;

BOOL rz=GetDiskFreeSpaceEx(FlashPath , &FreeBytes,&TotalBytes,&TotalFree);
MemTotal=TotalBytes.QuadPart;
MemFreeSpace=TotalFree.QuadPart;
MemFreeSpaceRate=((float)(MemFreeSpace)*100.0)/(float)MemTotal;

3.9 Retrieve MAC address

// useful for OEM using BSP

```
PIP_ADAPTER_INFO pAdapterInfo;
PIP_ADAPTER_INFO pAdapter;
ULONG ulOutBufLen = sizeof(IP_ADAPTER_INFO);

// Make an initial call to GetAdaptersInfo to get the necessary size
into the ulOutBufLen
pAdapterInfo = (IP_ADAPTER_INFO *) malloc(sizeof(IP_ADAPTER_INFO));
if (GetAdaptersInfo(pAdapterInfo, &ulOutBufLen) ==
ERROR_BUFFER_OVERFLOW)
{
free(pAdapterInfo);
pAdapterInfo = NULL;
pAdapterInfo = (IP_ADAPTER_INFO *) malloc(ulOutBufLen);
}

// Get adapter info
if (GetAdaptersInfo(pAdapterInfo, &ulOutBufLen) == NO_ERROR)
{
for (pAdapter = pAdapterInfo; pAdapter; pAdapter = pAdapter->Next)
{
if (pAdapter->Type == IP_ADAPTER_802_3) break;
}
}

// Cannot find the best adapter, use the first one we found
if (!pAdapter)
{
pAdapter = pAdapterInfo;
}

strcpy(szIpAddress, pAdapter->IpAddress);
// MAC address = AA:BB:CC:DD:EE:FF:GG (IPv4)
sprintf(szMacAddress, "%.2X%.2X%.2X%.2X%.2X%.2X",
pAdapter->MacAddress[0], pAdapter->MacAddress[1], pAdapter->MacAddress[2],
pAdapter->MacAddress[3], pAdapter->MacAddress[4], pAdapter->MacAddress[5]);

// Don't forget to free memory!
if (pAdapterInfo)
{
free(pAdapterInfo);
pAdapterInfo = NULL;
}
```

3.10 Call Backlight control dialog

```
typedef BOOL (FAR WINAPI *ADVANCEDPROC)(HWND);

int _tmain(int argc, _TCHAR* argv[])
{
    HINSTANCE hAdvanced = NULL;
    ADVANCEDPROC pfnAdvanced = NULL;

    if (hAdvanced = LoadLibrary(TEXT("MyBacklight.dll"))) {
        //Dll filename, load dll.
        pfnAdvanced = (ADVANCEDPROC)GetProcAddress(hAdvanced,
TEXT("BacklightAdvApplet")); //Function name, get function pointer.
    }
    if (pfnAdvanced) {
        BOOL ret = pfnAdvanced(NULL);
        //Run function, show dialog.
    }
    FreeLibrary(hAdvanced);

    return 0;
}
```

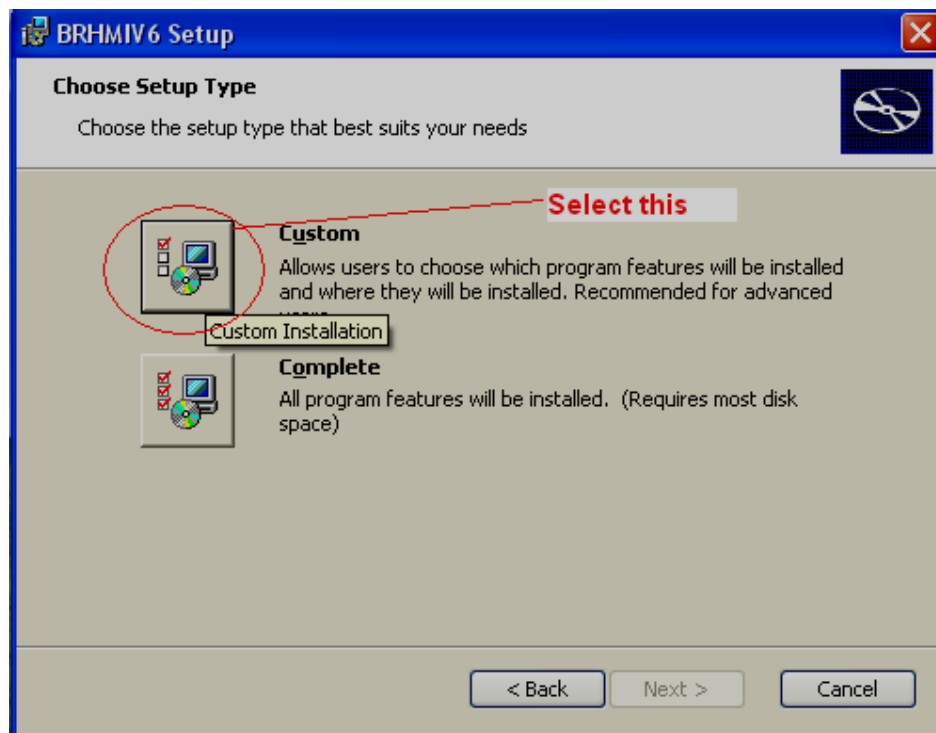

4. VISUAL STUDIO SAMPLE

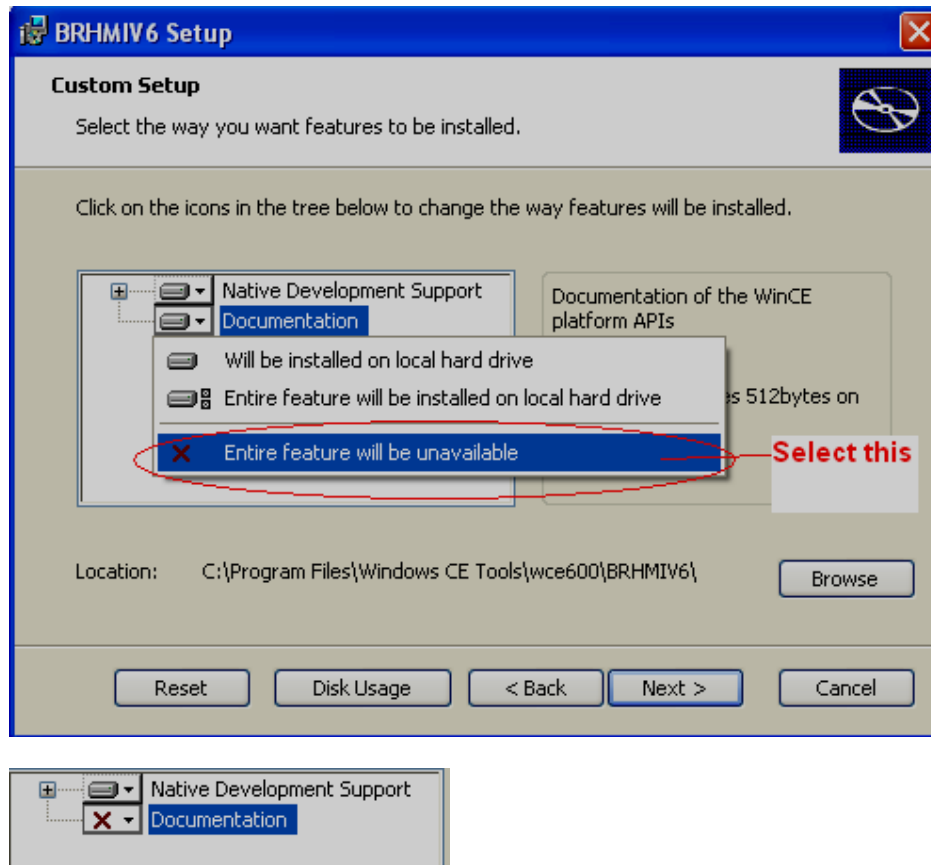
How to run Visual studio application in WinCE HMI

Step-1: Software installation

1. Install Visual studio 2008 professional edition
2. Update to Visual studio 2008 SP1
3. Install Windows embedded CE 6.0 software
4. Install SDK file. Contact factory for the SDK file. This file will be supplied in CD when you order HMI for OEM applications with software ordering code 0. Conditions apply to supply this file. Please contact factory for further details

If you get any error message during installation of SDK file, select custom install. Deselect documentation and then install again



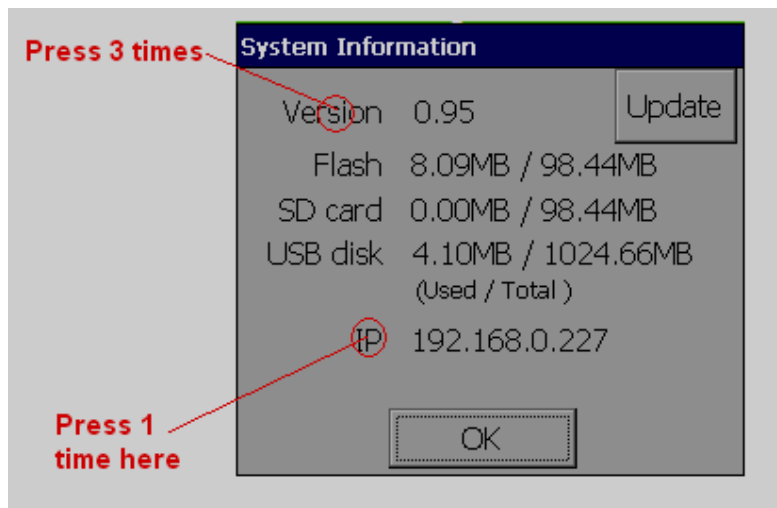


If still there are problems, please use Visual studio 2005 (Visual studio 2005 must be update to SP1)

5. Download Microsoft ActiveSync 4.5 from internet

Step-2: How to open Windows CE desktop in HMI

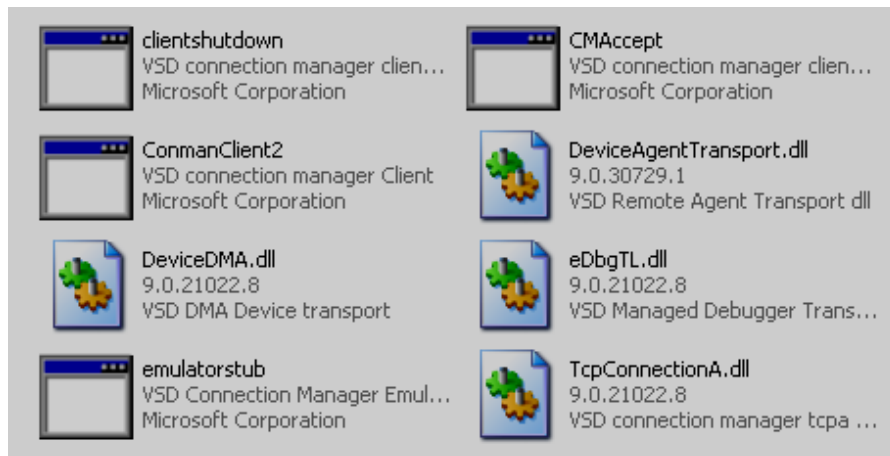
1. If "Control Center" firmware installed in HMI, at power on, it shows Control center screen
2. Press at "System information"
3. Press on top of version (middle area) 3 times quickly. Then, press on top of "IP" for 1 time. Then, it will close firmware and open "Windows CE desktop"



Note: If you order HMI for OEM application, if it is mentioned clearly while ordering the HMI, then, we will not load Windows CE image with “Control Center” firmware into HMI. On Power on, you can see Windows CE desktop directly

Step-3: Copy ARM files from PC to HMI

After installation of Visual studio 2008 software in your PC, search for the following files and place them in a folder by name ARMV4i



Generally, you may find these files in your PC after installation of Visual studio software at following location based on your installation drive

E:\Program Files\Common Files\Microsoft Shared\CoreCon\1.0\Target\wce400\armv4i

1. Open Windows CE desktop in HMI
2. Double click at “My Device”
3. Double click at “Residential flash”

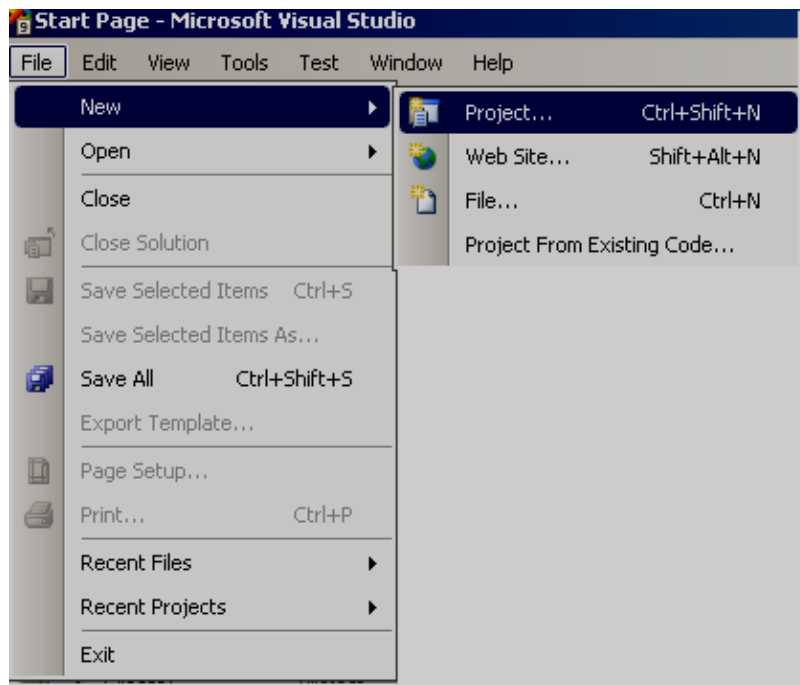
4. Copy folder ARM4Vi to “Residential flash”
5. Open folder ARM4Vi in HMI
6. Select file ConManClient2. Then, click at “Open” (Select this file for extended period to appear the new window to select Open)
7. Select file Cmaccept Then, click at “Open” (Select this file for extended period to appear the new window to select Open)

Note: If above two files are not running in HMI, you might be not able to download application from PC to HMI later

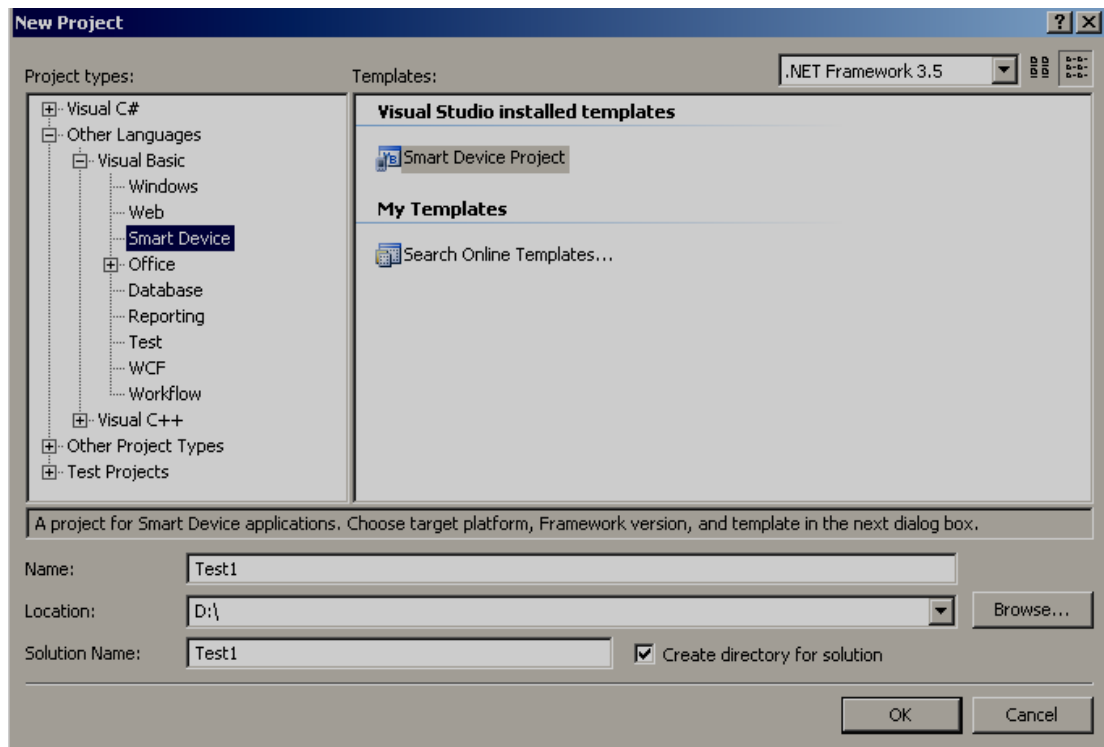
Step-4: Procedure to create a small application (Visual basic)

Open Visual studio 2008

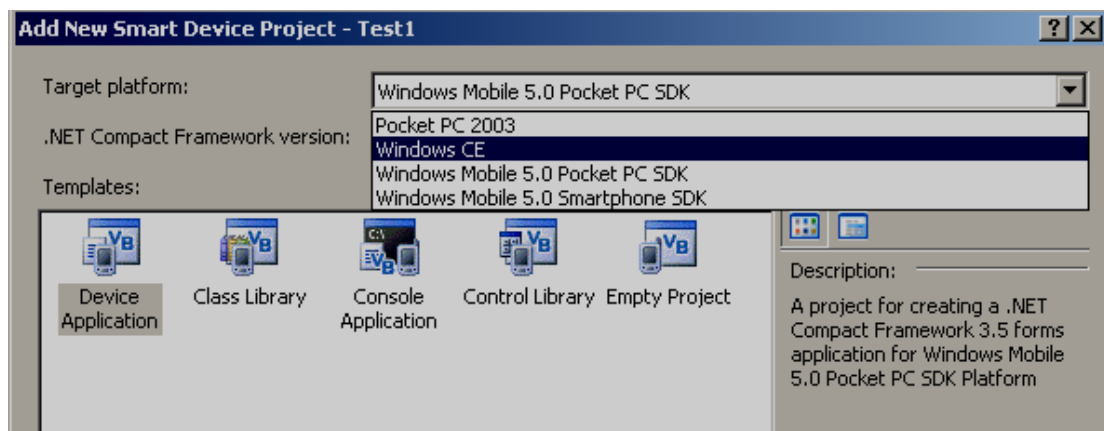
Create a new project



Select “Smart Device” under Visual basic, Enter project name say “Test1”, then click at “OK”

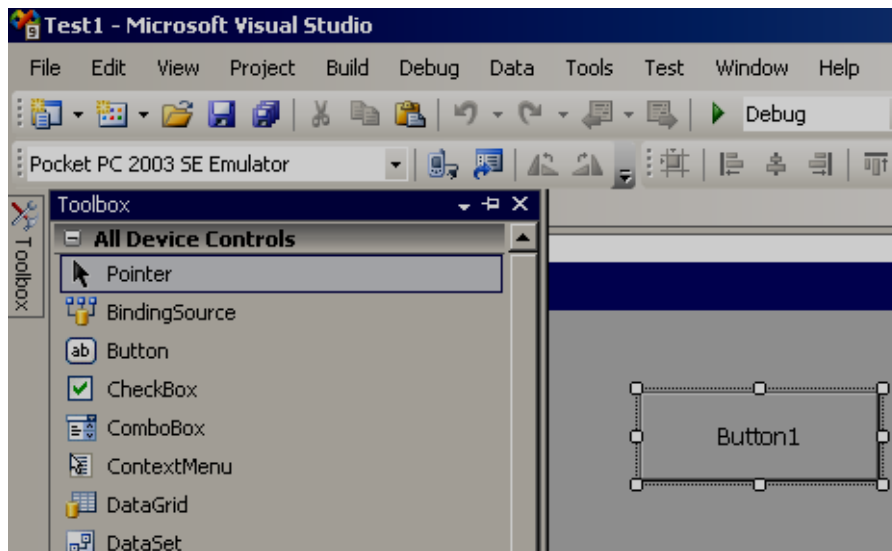


Select Target platform as “Windows CE”



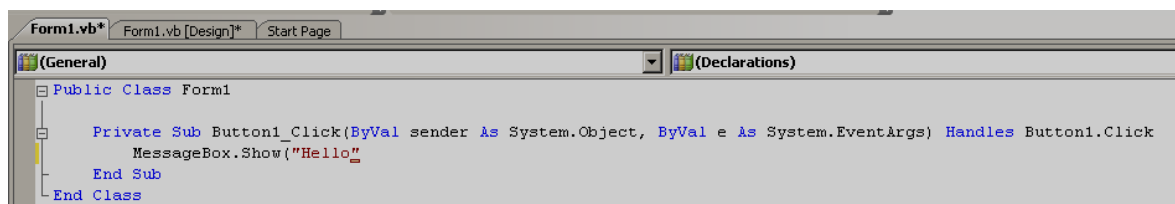
Select “Device Application”, click at “OK”

Insert a button from Tools box.



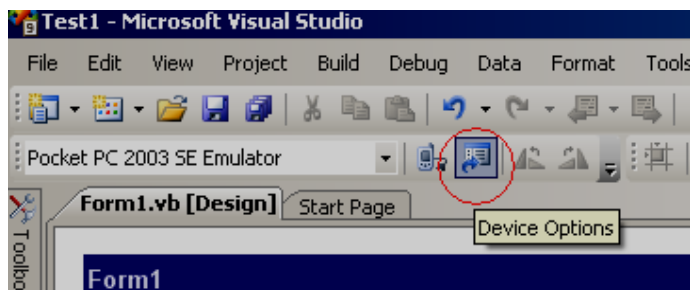
Double click on button and write a small program as below

```
MessageBox.Show("Hello")
```

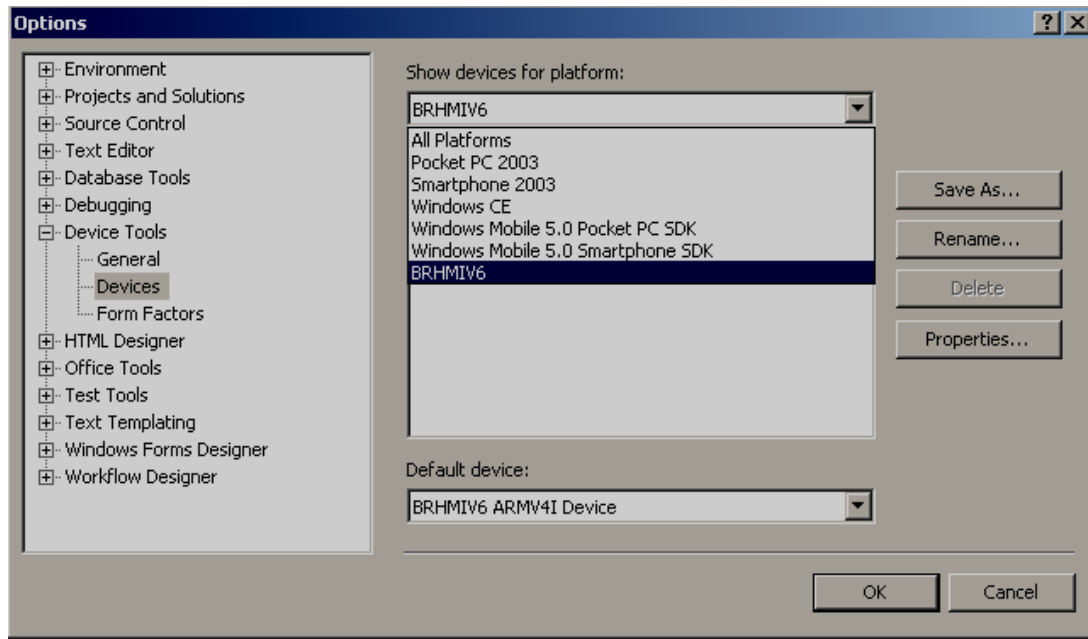


Save and close the above screen

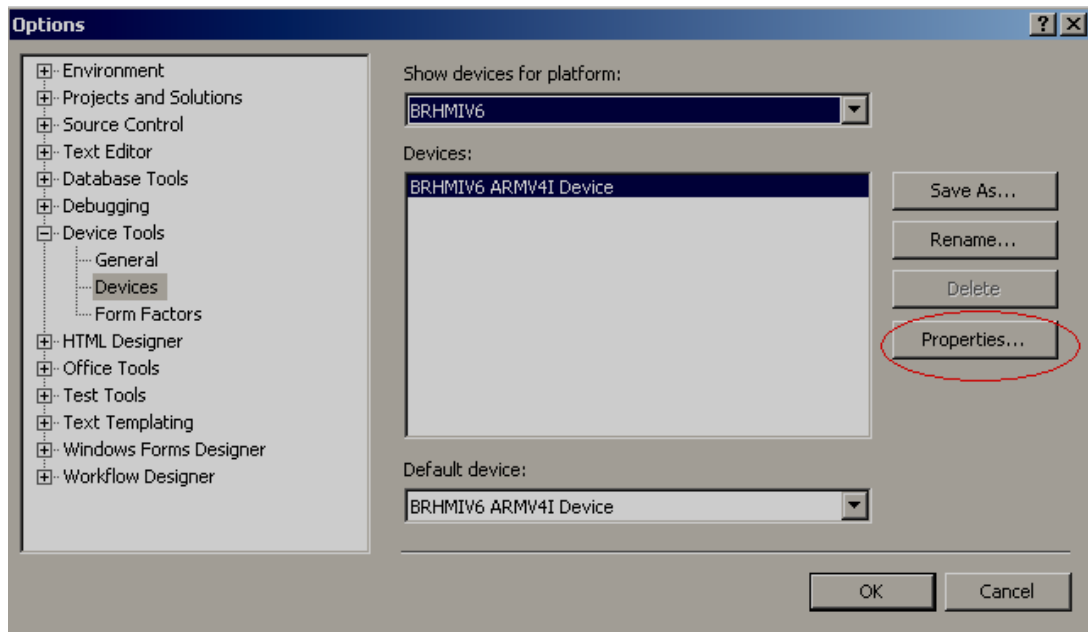
Click at Device Options



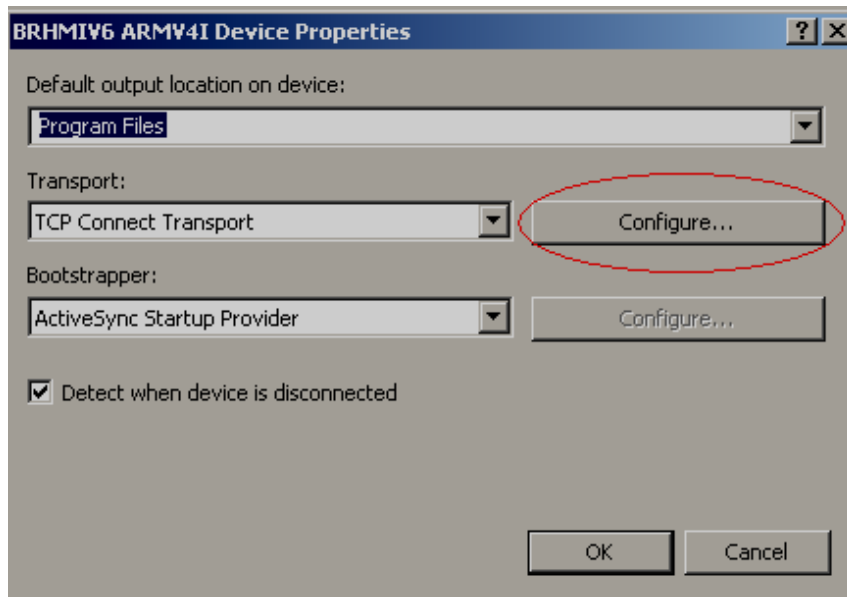
Select BRHMIV6



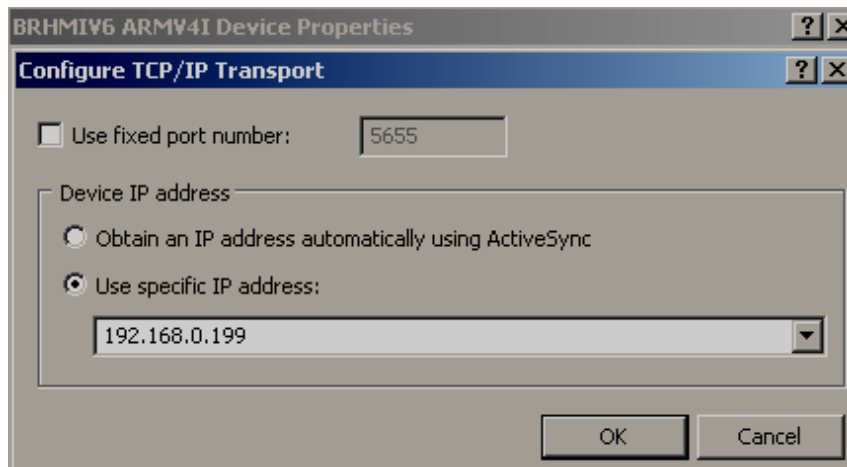
Click at “Properties”



Click at “Configure”

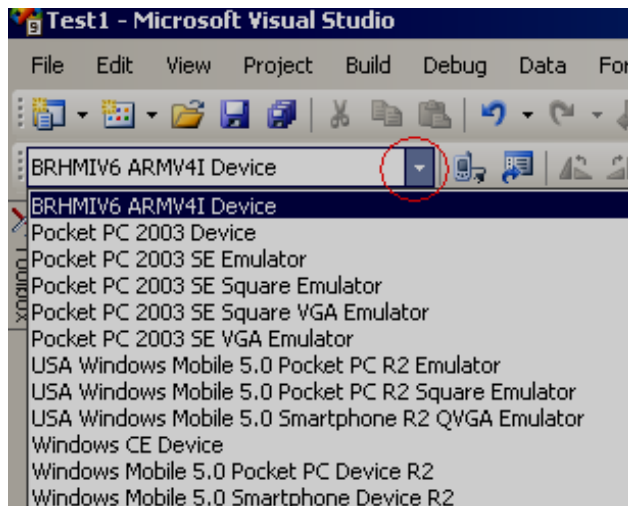


Enter IP address of HMI

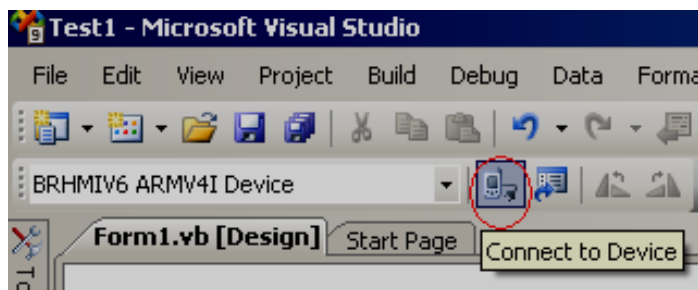


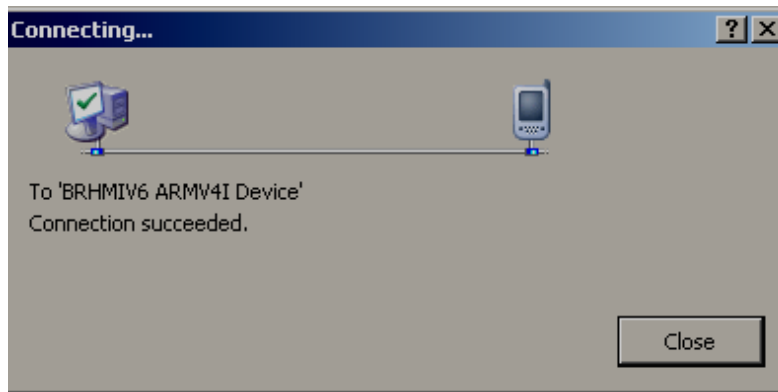
Click at "OK" and close all the screens

Select BRHMIV6



Click at "Connect to device" icon

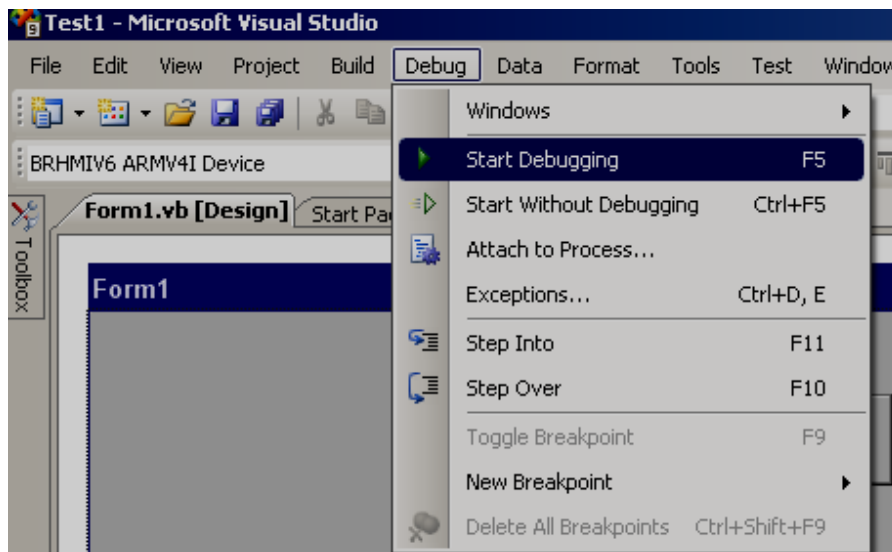


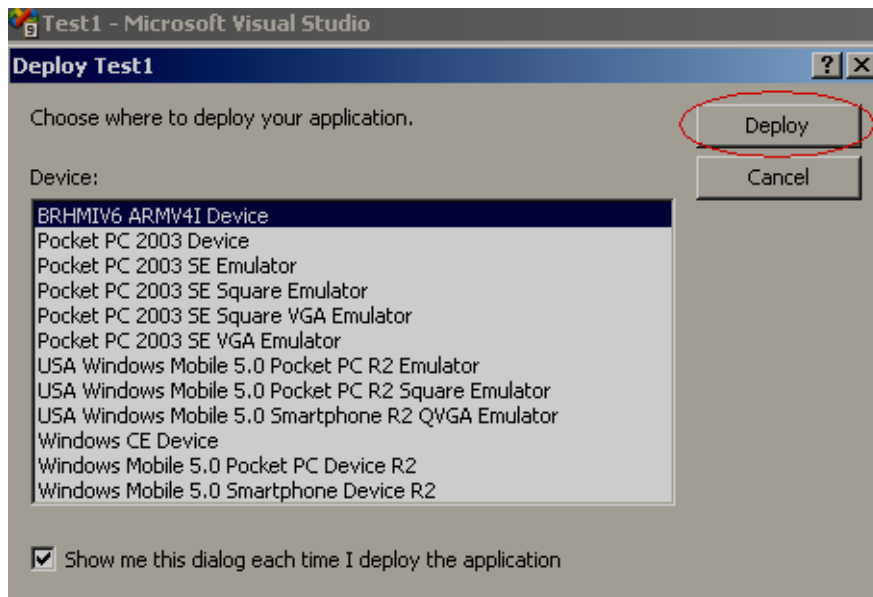


If connection is not succeeded, then, close the HMI screen. Make sure that ConManClient2 is open first and then open in correct sequence and try again

If connection is succeeded, then build project and deploy solution

Make sure there are no errors and rebuild succeeded message appears as above



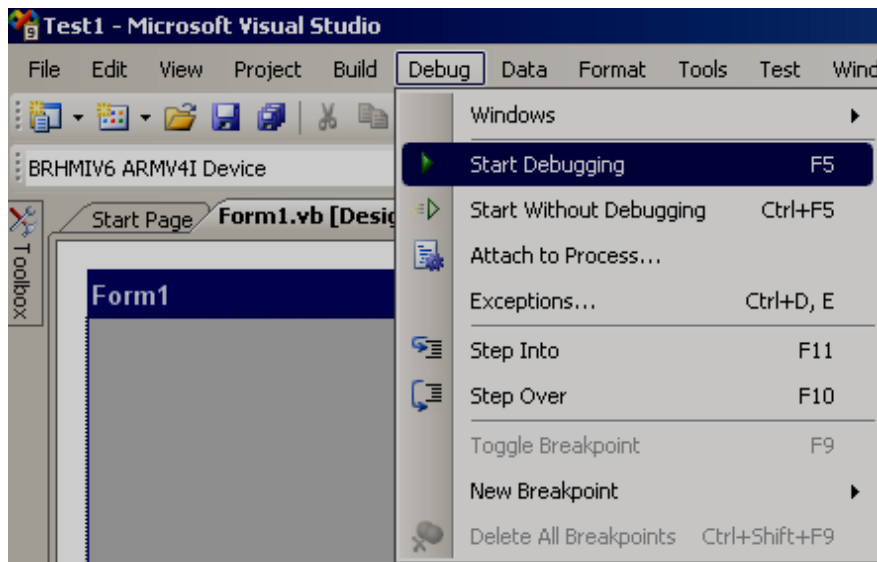


Select BRHMIV6 ARM4Vi device and click at “Deploy”

Now in HMI, check Program files available at “My devices”. There should be a folder with name Test1 (Same as project name defined earlier). Open Test1 folder and then run Test1.exe file

Press on Button. It will popup a screen with message “Hello”

If you wish to run above application directly from PC, then, start debugging or press F5



5. FAQ

1. I would like to run Indusoft/Wonderware/Movicon SCADA software in your HMI. Is it possible?

Yes, possible. You need to obtain WinCE run time files from the software provider and install them to run third party WinCE based SCADA in our HMI

2. Is it possible to run Visual studio applications in WinCE Core or WinCE Pro? Do you have any sample program for this? Do I need BSP or SDK for this?

Yes, possible. You need SDK for this. Please contact factory for Software development kit (SDK) files

You don't need BSP files for this purpose

3. Can you supply Board support package (BSP) for your HMI? What is the use of BSP?

Yes, please contact factory for BSP files

In case if OEM company have professional WinCE engineers and wish to customize WinCE image for their requirement, then, they need BSP.

4. Can you supply HMI with Linux?

Regret, it is not possible.

5. I would like to start some application automatically at HMI Power ON. Is it possible?

Yes, there is a tool included in WinCE image. It's name is "Auto Run Maker". You can find this at "Control Panel" in Windows. You can use Auto Run maker to select the path/folder/file to start automatically after Power ON. Please refer BSP documentation for more details.

6. Is it possible to convert WinCE Core to WinCE Pro image?

Yes, it is possible. Please contact factory for the correct WinCE image files. Please do note that, you need to buy WinCE Pro. License and replace the label on the HMI. You can use a tool "Upgrade System" available at "Control Panel" in windows for this.

7. Is it possible to install "Control Center" in Windows CE Professional version?

Regret, not possible

8. I would like to include my logo or show some special screen at HMI Power ON. Is it possible? Do I need SDK or BSP for this?

There is a tool available in our WinCE image "Auto Maker". Using this tool, it is possible to select boot Logo or Desktop screen. It's easy to do this and you don't need either SDK or BSP for using ready available tool

Please refer section "Auto Run Maker" for more details

If you have professional engineers familiar with WinCE, then, you can also do this using BSP files.

SDK is not useful to create boot up images

9. I have HMI with WinCE Core license. I need to convert to WinCE Pro. Can I buy license locally? What is the ordering code?

Yes, you can buy this locally if available.

884-00341-1PK MST

Windows CE Pro 6.0 EMB ESD OEI Pro Runtime

10. I want to adjust brightness in HMI. Is it possible?

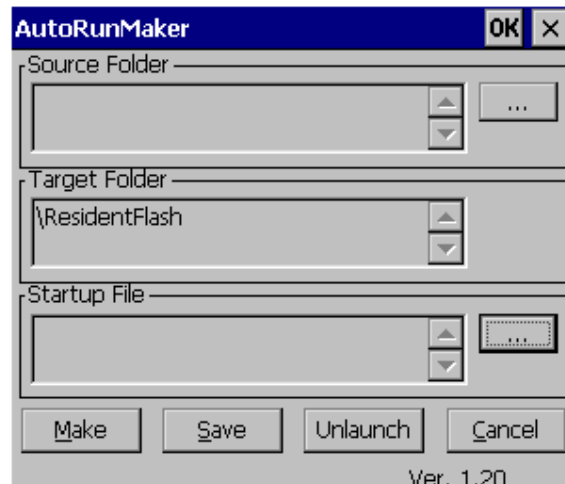
Yes, it is possible to adjust brightness. Tool available for this and please check the manual for more details

11. How I know the current build version for the WinCE image?

Open "My device"

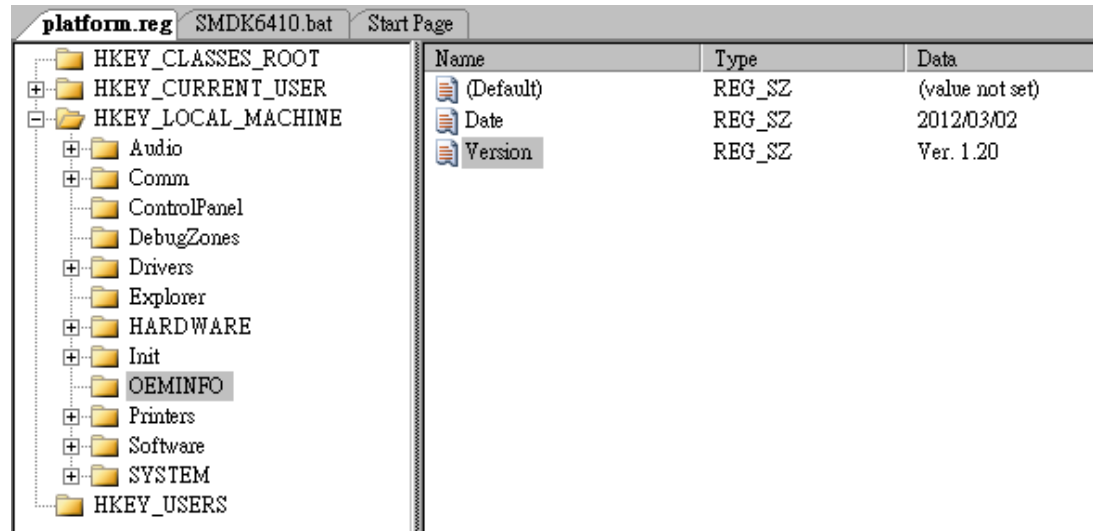
Open "Control Panel"

Open "Auto Run Maker"



12. I am using BSP, how I can know current version of BSP files?

It's possible to check this from platform builder by engineer familiar with WinCE

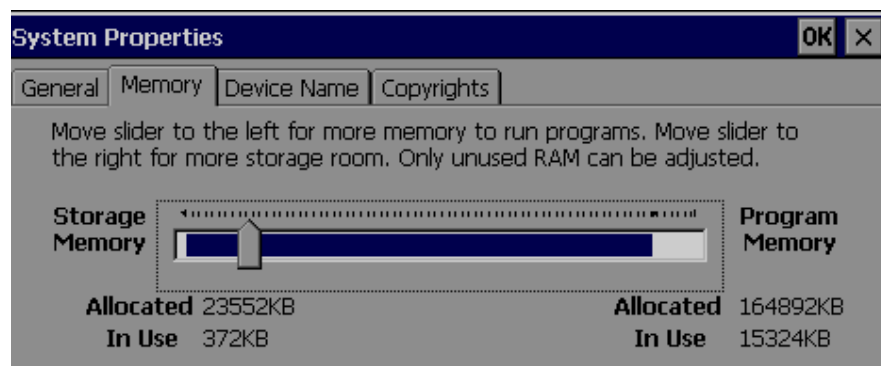


13. How to find Program memory left for applications? How to calculate this? Where I can find this information?

WinCE image will be loaded into RAM at Power ON. Some memory will be reserved for boot loading

For ex: HMI 7" (High end), RAM = 256 MB
WinCE Pro. Image size = 55 MB (Approx.)

Then system memory = 256 MB – 55 MB = 201 MB (approx.)



In above fig, system memory = Storage memory + Program memory

Allocated system memory = 23552 + 164892 = 188444 KB (Approx 184 MB)

Free Prog. Memory = Allocated Prog. Memory – Prog. Memory in use

In above example, $164892 - 15324 = 149568$ KB

Approx. 146 MB free



Please check HMI available with you to know the exact Program memory available for applications. The above calculation is for reference only